

Nazarov Evgeny Anatolyevich - Mikhailovskaya Military Artillery Academy; e-mail: era-nazarov@yandex.ru; St. Petersburg, Russia; phone: +79964061823; cand. of eng. sc.; doctoral researcher.

Danilin Maxim Evgenievich - JSC NPO Avanti; e-mail: info@avgr.tech; phone: +78001004194; Kazan, Russia; general director.

Kosenko Evgeny Yurievich - Research Institute of Robotics and Control Processes of the Southern Federal University; e-mail: ekosenko@sfedu.ru; Taganrog, Russia; phone: +79045071963; cand. of eng. sc.; senior research associate.

УДК 004.9+658.5

DOI 10.18522/2311-3103-2024-6-42-52

К.О. Обухов, И.Ю. Квятковская, А.В. Морозов

МЕТОД ГЕНЕТИЧЕСКОГО ПРОГРАММИРОВАНИЯ РЕШЕНИЯ ЗАДАЧИ ОПЕРАТИВНО-КАЛЕНДАРНОГО ПЛАНИРОВАНИЯ ДИСКРЕТНОГО ПРОИЗВОДСТВА

Одним из основных условий успешного функционирования предприятия является грамотно организованный процесс производственного планирования. Автоматизировать эту деятельность позволяют системы производственного планирования класса APS/MES, основой которых являются алгоритмы построения производственных планов. В работе исследуется проблема составления расписания для предприятий дискретного типа производства, относящаяся к области задач многокритериальной оптимизации. Приведено формальное описание задачи планирования с учетом основных производственных ограничений (временные ограничения, требования к оснастке и порядку выполнения операций). Кратко рассмотрены основные методы решения задач данного класса, отмечены их основные достоинства и недостатки. Для решения поставленной задачи выбран подход на основе генерации эвристических правил, применяемых при планировании производственных операций на заданные ресурсы. На основе данного подхода предложен двухэтапный алгоритм построения производственных расписаний, включающий в себя генерацию правил диспетчеризации и их дальнейшее применение при построении расписания. За генерацию правил диспетчеризации отвечает генетический алгоритм. Подробно описана реализация его генетических операторов, а также состав хромосомы и древовидное представление входящих в хромосому правил диспетчеризации. Реализация алгоритма выполнена на языке C# 12 с использованием свободной платформы .NET 8. Построенный алгоритм показал свою эффективность по сравнению с жадным алгоритмом на небольших сгенерированных наборах данных. Дальнейшими исследованиями в этой области является оценка эффективности построенного алгоритма с более сложными генетическими операторами и структурой дерева выражений, а также снижение длительности процесса генерации эвристических правил для больших наборов данных.

Задачи планирования; дискретное производство; теория расписаний; генетические алгоритмы; многокритериальная оптимизация.

K.O. Obukhov, I.Yu. Kvyatkovskaya, A.V. Morozov

METHOD OF GENETIC PROGRAMMING FOR SOLVING THE PROBLEM OF OPERATIONAL SCHEDULE PLANNING OF DISCRETE PRODUCTION

One of the main conditions for the successful functioning of the enterprise is a well-organized production planning process. Production planning systems of the APS/MES class, the basis of which are algorithms for building production plans, allow automating this activity. The paper examines the problem of scheduling for enterprises of a discrete type of production, related to the field of multi-criteria optimization problems. A formal description of the planning task is given, taking into account the main production constraints (time constraints, equipment requirements and the order of operations). The main methods of solving problems of this class are briefly considered; their main advantages and disadvantages are noted. To solve this problem, an approach based on the generation of heuristic rules used in planning production operations for specified resources has been chosen. Based on this approach, a two-stage algorithm for building production schedules is proposed, which includes the generation of dispatching rules and their further application in building schedules. A genetic algorithm is responsible for generating dispatch rules. The implementation of its genetic operators is described in detail, as well as the composition of the chro-

mosome and the tree representation of the dispatch rules included in the chromosome. The algorithm is implemented in C# 12 using a free platform.NET 8. The implemented algorithm has shown its effectiveness in comparison with the greedy algorithm on small generated datasets. Further research in this area is aimed at evaluating the effectiveness of the constructed algorithm with more complex genetic operators and the structure of the expression tree, as well as reducing the duration of the process of generating heuristic rules for large data sets.

Planning tasks; discrete production; scheduling theory; genetic algorithms; multi-criteria optimization.

Введение. Эффективное производственное планирование является одним из ключевых факторов успешной деятельности предприятия. Эта деятельность требует не только наличия подготовленных кадров в сфере управления производством, но и подходящих инструментов – таких как автоматизированные системы управления производством. Необходимость в разработке и внедрении отечественных систем производственного управления и планирования является очень важной задачей, актуальность которой подчеркивается и на государственном уровне [1, 2]. Это связано в первую очередь с тем, что большинство систем управления производством на отечественном рынке до 2022 г. имели иностранное происхождение (программные продукты от Siemens, Dassault и других зарубежных вендоров, относящиеся к системам класса APS/MES) [3]. Внедрение подобных автоматизированных систем должно увеличить количество производимой продукции, повысить эффективность использования оборудования и материалов [4, 5].

Основным компонентом любой системы планирования являются её алгоритмы, позволяющие построить приближенное к оптимальному расписание по некоторым критериям (как правило, заданными разработчиками самой системы). Большинство задач планирования относятся к NP-полным задачам дискретной оптимизации. К их числу относится задача построения оперативно-календарного плана для предприятий дискретного производства. Как правило, производственных процесс подобных предприятий характеризуется сложными технологическими процессами, требующими соблюдения различных производственных ограничений. К таким ограничениям относится соблюдение последовательности выполнения операций и требований к использованию оснастки. Критериями оптимальности могут быть как уменьшение просроченных заказов, так и снижение временных издержек (время наладки оборудования или его простоя).

Разработка математико-алгоритмического комплекса для решения задач построения расписания, применимого для различных предприятий дискретного производства, позволит повысить их эффективность путем минимизации выбранных критериев оптимальности, отличающихся для конкретных производств. Разработанные алгоритмы могут быть в дальнейшем использованы в тиражной автоматизированной системе, которая может быть развернута для различных предприятий дискретного производства. Таким образом, одним из основных критериев к разработанным алгоритмам и их реализации является их универсальность и возможность их настройки путем указания различных метрик и критериев оптимальности. Для достижения поставленных целей необходимо формализовать задачу построения расписания для предприятий дискретного производства с учетом основных производственных ограничений, рассмотреть различные методы, используемые для решения задач планирования, разработать алгоритм планирования и произвести его апробацию.

1. Постановка задачи планирования дискретного производства. Систематизация и решение задач планирования рассматривается в рамках такого раздела дискретной математики, как теория расписаний [6]. Процесс планирования состоит из назначения работы (job) или конкретных операций работы некоторым исполнителям, чаще всего называемым машинами (machine) В отечественной и зарубежной литературе выделяют несколько подклассов задач теории расписаний [7], основными из которых являются:

1. Flow-Shop Scheduling (задачи конвейерного типа). В таких задачах работы представляют собой последовательность одних и тех же операций в одном и том же порядке, при этом расписание каждой машины представляет собой описание порядка выполнения операций каждой из работ. При этом каждая работа должна быть обработана на всех машинах (каждой операции задана конкретная машина).

2. Open-Shop Scheduling (открытая линия). Схожи с предыдущим классом задач, но не имеют ограничений на порядок выполнения операций каждой работы.

3. Job-Shop Scheduling (рабочий цех). Для каждой из работ указано отношение предшествования операций (в некоторых случаях это отношение может быть определено и между отдельными работами), также для каждой работы (или конкретных операций) задано множество машин, которые могут её выполнять.

Рассматриваемая задача построения расписания ближе всего относится к последнему классу – Job-Shop Scheduling Problem (JSSP), и описывается следующим образом. Задается набор производственных заказов (работы в терминологии JSSP) *Orders*. Каждый элемент *order_i* представляет собой кортеж $\{s_i, d_i, process_i, prevOrd_i\}$, где s_i – дата начала заказа, d_i – дата окончания, $prevOrd_i \subset Orders$ – список предшествующих к выполнению заданий, $process_i$ – техпроцесс заказа, $process_i = \{O_{i1}, \dots, O_{ij}\}$. Каждая j -я операция i -го заказа представляется в виде кортежа $O_{ij} = \{p_{ij}, MC_{ij}, prevOp_{ij}\}$, где p_{ij} – нормативное время выполнения операции, MC_{ij} – набор альтернативных комбинаций машин (ресурсов) для ее выполнения, $MC_{ijk} \subset M$, $prevOp_{ij} \subset process_i$ – набор предшествующих к выполнению операций. В комбинацию ресурсов может входить основное оборудование (станок), требуемый персонал и оснастка (если имеются ограничения в её работе на нескольких станках).

Как и в классической постановке задачи JSSP, задается набор машин $M = \{m_1, m_2, \dots, m_M\}$. Каждая машина может обрабатывать только одну операцию в момент времени t . Для каждой $m \in M$ задана функция доступности $Availability(m, t) \rightarrow \begin{cases} 0, & \text{машина занята} \\ 1, & \text{машина } m \text{ доступна в момент времени } t \end{cases}$, описывающая доступность машины согласно графику доступности. Также задана функция расчёта переналадок $Adj(m, o_{prev}, o_{next}) \rightarrow \mathbb{N}_0$, возвращающая длительность переналадки (в секундах) для операции o_{next} на заданной машине m и предыдущей операции o_{prev} , $o_{prev}, o_{next} \in \bar{O}$, где $\bar{O} = process_1 \cup process_2 \cup \dots \cup process_n$. Переналадка означает время, необходимое для подготовки операции к выполнению на заданных машинах.

Описанная модель переналадок относится к переналадкам, зависящим от последовательности операций (sequence-dependent setup times) [8]. В системах производственного планирования для её описания чаще всего используют представление переналадок в виде матрицы с указанием нормативного времени наладки [9], но в некоторых случаях нормативное время неизвестно, либо оно изменяется при определённых внешних условиях.

Таким образом, решаемая задача является расширением классической задачи Job Shop Scheduling Problem с некоторыми дополнительными ограничениями, которые свойственны этапу оперативного планирования для дискретного производства – учет дополнительных ресурсов и оснастки, наладки оборудования и учет их календарной доступности. Наиболее близкий подкласс задач – DFJSS (Dynamic Flexible Job Shop Scheduling), описывающий задачи планирования с возможностью выбора машины для каждой операции, при этом время поступления работ распределено случайным образом. Для таких задач характерно наличие двух процессов принятия решения – принятие решения о выборе следующей операции к планированию из множества доступных, а также выбор подходящей машины (комбинации машин) для выбранной операции (в зарубежной литературе – sequencing / routing problems).

Как правило, критерий оптимизации описывается в виде максимизации или минимизации функции вида $f(s)$, где $s \in S$ – расписание из множества возможных расписаний S . Таким критерием может быть длительность переналадок или количество выполненных в срок заказов. Основная сложность возникает в том случае, если критериев оптимизации несколько. Это может произойти по следующим причинам:

1. Изменилась основная цель планирования.
2. Изначальную цель планирования невозможно (или очень сложно) представить в виде одного критерия.

Первый случай может произойти при определенных изменениях на производстве (например, появился новый срочный заказ на производство или нарушилась цепочка поставок, из-за чего более приоритетным становится учет сроков выпуска заказов, а не минимизация издержек). Повторный перерасчет расписания с последующим его анализом и внесением изменений может быть очень трудоемкой задачей даже при наличии средств автоматизированного планирования. При этом нужно учесть, что часть производственного задания уже может быть взята в работу.

Второй случай характерен при наличии нескольких критериев с одинаковым или различным приоритетом – например, оптимальным считается план с максимальным количеством выполненных в срок заказов и с минимумом издержек или переналадок. Эти параметры могут иметь различные единицы измерения и по-разному интерпретироваться, из-за чего их сложно «свернуть» в один. Кроме того, порядок применения критериев также выбирается ситуационно.

Стоит отметить, что большое количество систем производственного планирования имеет фиксированное количество критериев оптимизации, выбираемых пользователем. При расчете плана может использоваться только один критерий, либо уже определенная комбинация (при выборе определенного режима планирования). Добавление новых критериев при этом, как правило, требует разработки нового алгоритма планирования под конкретное производство.

Таким образом, описанная задача производственного планирования переходит в разряд задач многокритериальной оптимизации (МКО).

2. Краткий обзор различных методов, применимых для решения задач построения расписаний. Как правило, методы решения задач теории расписаний разделяются на несколько основных групп. Точные методы (полный перебор, метод ветвей и границ [10], динамическое программирование [11]) малоприменимы из-за высокой вычислительной сложности, и подходят только для очень небольшого размера задачи (до десятка машин/работ).

Более простым (и, чаще всего, приближенным к реальности) является использование жадных алгоритмов и различных правил диспетчеризации [12]. Данные правила описывают логику выбора операций или машин к планированию. В описываемой задаче производственного планирования дискретного производства, относящейся к классу DFJSSP (Dynamic Flexible Job Shop Scheduling Problem) можно выделить два правила для принятия решений:

1. Sequencing – выбор следующей операции к планированию из доступных на данный момент.
2. Routing – выбор комбинации машин, на которой будет запланирована выбранная операция.

Общая схема пооперационного планирования при таком подходе можно представить в виде блок-схемы (рис. 1).

Таким образом, процесс планирования является циклическим, на каждой из его итераций принимаются описанные выше решения. Функции $Routing(o)$ и $Sequencing(o, m)$ описывают расчёт приоритета согласно правилам диспетчеризации соответствующего вида, $o \in \bar{O}$, $m \in M$. Планирование происходит слева-направо, от дат запуска производственных заказов.

К минусам правил диспетчеризации можно отнести их слабую универсальность – правила, используемые одними предприятиями, могут не подходить для других, также они могут меняться и в рамках одного предприятия, например, в зависимости от загруженности производственными заказами.

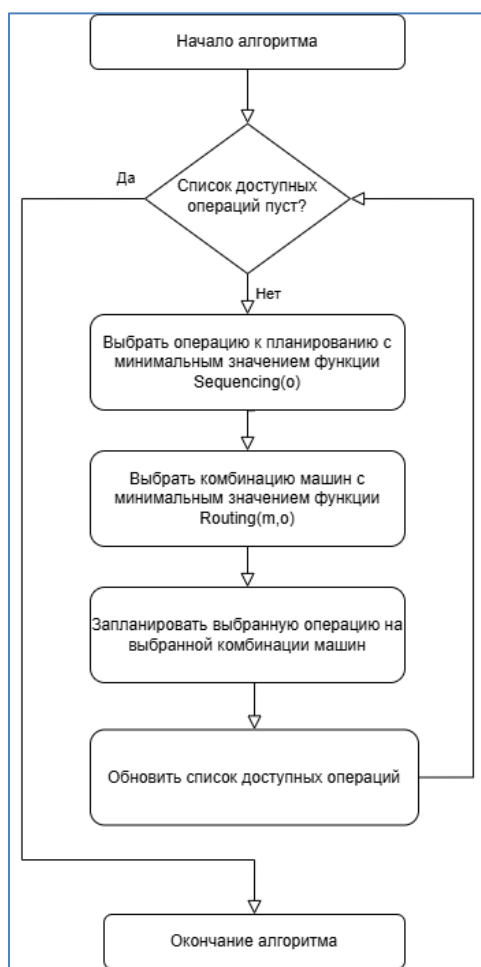


Рис. 1. Блок-схема планирования с учетом правил диспетчеризации

Метаэвристические методы (в частности, эволюционные алгоритмы) позволяют получить приближенное к оптимальному решение за относительно небольшое время [13–15]. К минусам можно отнести возможность попадания в локальный оптимум. Другая возможная проблема – сложность процедуры кодирования/декодирования решения, что особенно актуально для генетических алгоритмов. Например, при решении задач планирования при помощи генетических алгоритмов хромосому чаще всего представляют в виде последовательности операций, для каждой из которых сохраняются номера машин, которые будут её обрабатывать, и порядок обработки операции. Для получения информации по построенному расписанию (значения различных метрик) необходимо на основе этих данных построить расписание с учетом всех производственных и временных ограничений, а затем на основе полученного расписания произвести расчет метрик. Для большого набора данных этот процесс может быть достаточно трудоемким. Другой способ – модифицировать текущее расписание, не перестраивая его полностью – также может быть очень трудоемким, так как проверка корректности расписания с учетом описанных ограничений является более ресурсозатратным и сложным в реализации процессом.

Дальнейшим развитием описанных выше методов можно считать использование гиперэвристических (hyper-heuristics) методов для генерации эвристических правил. Суть подобных методов состоит в генерации некоторых эвристических правил, которые позволяют найти решение исходной задачи. Данный подход позволяет генерировать правила диспетчеризации в зависимости от текущей производственной ситуации. В контек-

сте использования генетических алгоритмов этот подход называют генетическим программированием. В отличие от описанного выше примера с представлением решения, хромосомой для такого алгоритма будет не само расписание, а правила диспетчеризации, необходимые для его генерации. Проблема генерации правил диспетчеризации широко освещалась в работах [16–18].

3. Описание алгоритма на основе генетического программирования. В рамках данной работы в качестве рассматриваемого метода был выбран метод генетического программирования. Несмотря на свою относительную новизну, данный подход показывает хорошие результаты при решении различных задач [19], в том числе задач построения расписаний. Общая схема работы алгоритма состоит из трех основных шагов – инициализации, генерации эвристики и построении расписания. Общая схема работы алгоритма представлена на рис. 2 в виде диаграммы потоков данных (DFD).

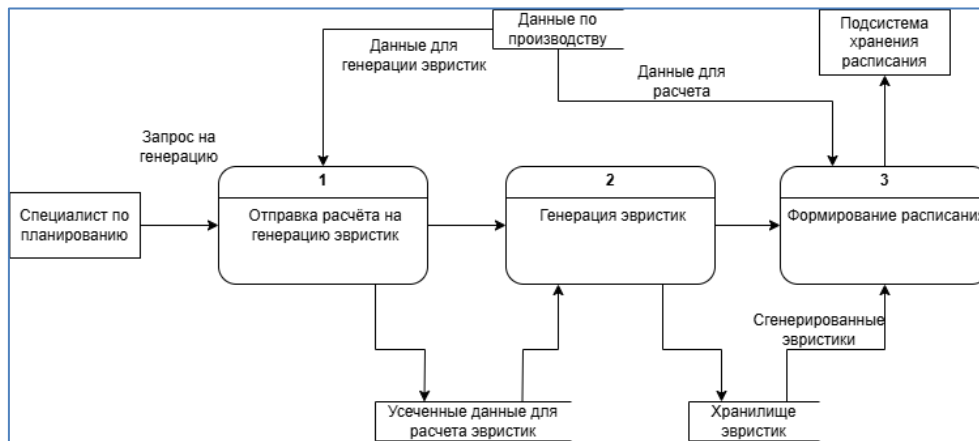


Рис. 2. Диаграмма потоков данных для разработанного алгоритма

Инициализация алгоритма включает в себя задание входных данных (списка задач и машин), а также одного или нескольких критериев оптимизации в порядке их приоритета. На основе этих данных происходит генерация эвристик планирования. Под эвристиками подразумеваются правила диспетчеризации для выбора операции (routing) и комбинации машин (sequencing). Сгенерированные эвристики сохраняются в виде дерева в отдельное хранилище с целью их дальнейшего использования при формировании расписания. Представление в виде дерева было выбрано по причине удобства его модификации и интерпретации, однако существуют и другие способы (например, представление в виде текста или низкоуровневого кода) [20]. Стоит учитывать, что данные для генерации эвристик, и данные, используемые при расчетах расписания на их основе, могут быть различными. Достоинством при этом может быть то, что более трудоемкий процесс генерации эвристик не вызывается каждый раз при каждом расчете расписания (для больших наборов данных это может быть очень трудоемкой операцией). Недостатком является то, что при сильной смене производственной ситуации (например, изменении номенклатуры выпускаемых изделий) старые эвристические правила могут стать неактуальными, и специалисту по планированию нужно будет принять решение о генерации новых эвристических правил.

Основным шагом алгоритма является этап генерации эвристик, основанные на уже упоминаемом ранее методе генетического программирования, основанным на классических генетических алгоритмах. На вход алгоритму, помимо данных о планируемых работах и машинах, передается набор критериев оптимальности, отсортированный в порядке их приоритета (важности). При декодировании каждой хромосомы происходит расчет расписания по схеме, описанной на рис. 1, после чего рассчитывается вектор чисел – значений каждого из критериев. Сравнение критериев происходит попарно, слева-направо (по аналогии с лексикографическим сравнением у строк).

Каждая хромосома представляет собой математическое выражение для каждого из правил (Routing/Sequencing). Выражение описывает функцию расчета приоритета для заданной операции/комбинации машин.

Выражения представляются в виде дерева. Нетерминальными узлами являются:

- ◆ Математические бинарные операции: +, -, *, /.
- ◆ Функции $\max(a,b)$, $\min(a,b)$.
- ◆ Значение по модулю (единственная на данный момент унарная операция).

Терминальные узлы дерева – различные метрики, рассчитываемые на основании конкретной комбинации машин или операции к выбору. Список рассматриваемых метрик приведен в табл. 1.

Таблица 1

Описание терминальных узлов деревьев выражений

Тег	Описание	Применимость в правилах	
		Routing	Sequencing
OD	Длительность рассматриваемой операции (часы)	+	+
OST	Время наладки рассматриваемой операции на рассматриваемой комбинации машин (часы)	+	-
OTD	Время, оставшееся до нормативной даты окончания при планировании рассматриваемой операции на рассматриваемой комбинации машин (routing) либо минимальное среди всех возможных комбинаций (sequencing) (часы)	+	+
JD	Длительность работы, в которую входит рассматриваемая операция (часы)	+	+
MD	Максимальное время простоя при планировании рассматриваемой операции на комбинацию машин (часы)	+	-
OCS	Расчетное время начала операции при планировании для рассматриваемой комбинации машин (часы, отступ от даты старта планирования)	+	-

Для генетического алгоритма были реализованы следующие генетические операторы:

1. Кроссовер – операция основана на обмене поддеревьями выражений (tree-swapping crossover). Для каждого из деревьев выражений выбираются случайные точки (узлы) скрещивания, дочерние узлы которых меняются местами для каждого из поддеревьев. Процедура выполняется для каждого из правил, следовательно, её результатом будут 4 отличающихся хромосомы.

2. Мутация – реализовано два вида мутаций. Первый – изменение части одного из правил на случайное поддерево. Второй – изменение случайного терминального узла на другой терминальный узел. Вторая мутация применяется в том случае, когда дерево выражения становится слишком большим (более 10 узлов).

3. Выбор родителей для скрещивания – классическая турнирная выборка (tournament selection). Этот алгоритм основан на цикле, в котором происходит случайный выбор хромосомы из популяции, и её сравнение с лучшим из выбранных ранее вариантов. Количество итераций – фиксированное число, равное 5.

Основные параметры расчета:

- ◆ Размер популяции: 1024 особи.

- ◆ Число поколений: 32.
- ◆ Число элитных хромосом: 8 (1/128 от популяции).
- ◆ Начальная популяция генерируется случайным образом, каждое эвристическое правило содержит от 1 до 3 нетерминальных узлов.
- ◆ Вероятность мутации: 0.2.

В качестве критериев оптимизации было выбрано 2 функции. Первой по приоритету является длительность производственных переналадок, второй – количество работ, не выполненных в срок.

Разработанный алгоритм реализован на языке программирования C# с использованием свободной платформы .NET 8.0. Тестовые прогоны осуществлялись на сгенерированных наборах данных небольшого размера (до 50 работ и 3 машин). Критерии оптимальности в порядке приоритета – суммарная длительность переналадок и количество незапланированных в срок работ. Описание тестовых прогонов и их сравнение с результатами работы жадного алгоритма приведено в табл. 2.

Таблица 2

Результаты тестовых запусков алгоритма

№ за-пуска	1-й критерий по приоритету (переналадки, секунды)			2-й критерий по приоритету (нарушение сроков, кол-во работ)			Длительность расчета, секунды (для генетического алгоритма)
	Жадный алгоритм	Генетическое программирование	Улучшение в %	Жадный алгоритм	Генетическое программирование	Улучшение в %	
1	79200	54000	31,82	38	29	23,68	75
2	72000	39600	45	38	29	23,68	69,36
3	79200	46800	40,91	44	30	31,82	78,55
4	90000	54000	40	42	29	30,95	73,90
5	97200	54000	44,44	40	33	17,5	65,57
6	93600	50400	46,15	41	25	39,02	73,84
7	57600	39600	31,25	35	26	25,71	43,93
8	75600	43200	42,86	43	37	13,95	76,93
9	97200	54000	44,44	41	28	31,71	86,75
10	86400	54000	37,5	37	25	32,43	70,70

При сравнении с классическими жадными алгоритмами решения на основе сгенерированных эвристик были эффективнее по переналадкам и выполненным в срок работам вплоть до 30-40% (на небольших и относительно простых наборах данных). Анализ сгенерированных правил диспетчеризации показал, что большинство «правильных» с точки зрения алгоритма правил чаще остальных используют операции $\max$ и $\min$, например, следующим образом: $\max(\min(MD, JD), \max(OCS, MD)) \cdot OST$. Стоит отметить, что сгенерированные правила часто могут требовать упрощения, например, как сгенерированное выражение $OST + \min(MD, MD)$.

Выводы. В результате проделанной работы разработан и описан алгоритм оперативного планирования дискретного производства, основанный на генерации эвристических правил выбора операций и комбинаций машин посредством метода генетического программирования. Опытным путём доказана эффективность его работы по сравнению с классическими жадными алгоритмами. В частности, значения критериев оптимальности на построенных расписаниях уменьшились на 30-40% по сравнению с расписаниями, построенными жадным алгоритмом на одних и тех же наборах данных. К достоинствам алгоритма можно отнести его универсальность (можно использовать различные критерии оптимальности, а также различные узлы для описания деревьев эвристических правил). Недостатки алгоритма – высокая длительность расчёта.

Дальнейшими улучшениями алгоритма могут быть добавление новых видов узлов для описания деревьев выражений и оценка эффективности полученных на их основе решений, а также применение более продвинутых генетических операторов. Другая важная задача – построение усеченного набора данных, который бы максимально соответствовал исходному набору данных с целью уменьшения длительности расчёта эвристик.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Стратегия цифровой трансформации обрабатывающих отраслей промышленности в целях достижения их «цифровой зрелости» до 2024 года и на период до 2030 года. – URL: https://www.tadviser.ru/images/8/83/Stateg_info_2021_compressed.pdf (дата обращения: 07.11.2024).
2. Корчевская А.А. [и др.]. Тренды и сценарии развития рынка систем расширенного планирования производства (APS-систем) в рамках направления «Технет» НТИ. Экспертно-аналитический доклад: монография. – СПб.: ПОЛИТЕХ-ПРЕСС, 2024. – 112 с.
3. Обухов К.О. Сравнительный анализ систем производственного планирования в разрезе работы с альтернативными цепочками производственных операций // Молодой ученый. – 2022. – № 20 (415). – С. 143-147. – URL: <https://moluch.ru/archive/415/91773/> (дата обращения: 07.11.2024).
4. Климова Е.З., Павлова И.А., Макушева Ю.А. Место и роль оперативно-производственного планирования в управлении предприятием // ЕГИ. – 2022. – № 42 (4). – URL: <https://cyberleninka.ru/article/n/mesto-i-rol-operativno-proizvodstvennogo-planirovaniya-v-upravlenii-predpriyatii> (дата обращения: 07.11.2024).
5. Джуха В.М., Кириллов Д.О. Потенциальные преимущества APS-систем в продажах и операционном планировании // УЭКС. – 2013. – № 7 (55). – URL: <https://cyberleninka.ru/article/n/potentsialnye-preimuschestva-aps-sistem-v-prodazhah-i-operatsionnom-planirovanii-1> (дата обращения: 10.12.2024).
6. Лазарев А.А., Гафаров Е.Р. Теория расписаний. Задачи и алгоритмы. – М.: Физический факультет МГУ, 2011. – 222 с.
7. Job shop scheduling: Classification, constraints and objective functions. – URL: https://www.researchgate.net/publication/339128833_Job_shop_scheduling_Classification_constraints_and_objective_functions (дата обращения: 07.11.2024).
8. A review on job shop scheduling with setup times. – URL: https://www.researchgate.net/publication/275317984_A_review_on_job_shop_scheduling_with_setup_times (дата обращения: 10.12.2024).
9. Планирование и оптимизация процессов переналадки. – URL: <https://habr.com/ru/companies/ds/articles/533788> (дата обращения: 10.12.2024).
10. A Flexible Branch and Bound Method for the Job Shop Scheduling Problem. – URL: https://www.researchgate.net/publication/238723625_A_Flexible_Branch_and_Bound_Method_for_the_Job_Shop_Scheduling_Problem (дата обращения: 07.11.2024).
11. Dynamic Programming for Routing and Scheduling. – URL: <https://research.vu.nl/ws/portalfiles/portal/42163578/complete+dissertation.pdf> (дата обращения: 07.11.2024).
12. Designing Dispatching Rules to Minimize Total Tardiness. – URL: https://www.researchgate.net/publication/226253624_Designing_Dispatching_Rules_to_Minimize_Total_Tardiness (дата обращения: 07.11.2024).
13. A Job-Shop Scheduling Problem (JSSP) using Genetic Algorithm (GA). – URL: https://www.researchgate.net/publication/242081421_A_JOB-SHOP_SCHEDULING_PROBLEM_JSSP_USING_GENETIC_ALGORITHM_GA (дата обращения: 07.11.2024).
14. Сочнев А.Н. Планирование производства на основе управляемой генетическим алгоритмом имитационной модели // Журнал СФУ. Техника и технологии. – 2021. – № 2. – URL: <https://cyberleninka.ru/article/n/planirovanie-proizvodstva-na-osnove-upravlyaemoj-geneticheskim-algoritmom-imitatsionnoy-modeli> (дата обращения: 07.11.2024).
15. Solving the Job Shop Scheduling Problem with Ant Colony Optimization. – URL: https://www.researchgate.net/publication/363500511_Solving_the_Job_Shop_Scheduling_Problem_with_Ant_Colony_Optimization (дата обращения: 07.11.2024).
16. Zhang F., Mei Y., Nguyen S. and Zhang M. Survey on Genetic Programming and Machine Learning Techniques for Heuristic Design in Job Shop Scheduling // in IEEE Transactions on Evolutionary Computation. – Feb. 2024. – Vol. 28, No. 1. – P. 147-167.
17. Dynamic flexible job shop scheduling method based on improved gene expression programming. – URL: https://www.researchgate.net/publication/343612768_Dynamic_flexible_job_shop_scheduling_method_based_on_improved_gene_expression_programming (дата обращения: 10.12.2024).

18. Zhang F., Mei Y., Nguyen S. and Zhang M. Evolving Scheduling Heuristics via Genetic Programming With Feature Selection in Dynamic Flexible Job-Shop Scheduling // in IEEE Transactions on Cybernetics. – April 2021. – Vol. 51, No. 4. – P. 1797-1811,
19. Genetic Programming: An Introduction and Tutorial. – URL: https://wiki.eecs.yorku.ca/course_archive/2010-11/W/4403/_media/gp1.pdf (дата обращения: 07.11.2024).
20. Поликарпова Н.И., Точилин В.Н. Применение генетического программирования для реализации систем со сложным поведением // Научно-технический вестник информационных технологий, механики и оптики. – 2007. – № 39. – URL: <https://cyberleninka.ru/article/n/primenenie-geneticheskogo-programmirovaniya-dlya-realizatsii-sistem-so-slozhnym-povedeniem> (дата обращения: 07.11.2024).

REFERENCES

1. Strategiya tsifrovoy transformatsii obrabatyvayushchikh otrasley promyshlennosti v tselyakh dostizheniya ikh «tsifrovoy zrelosti» do 2024 goda i na period do 2030 goda [Strategy for digital transformation of manufacturing industries in order to achieve their “digital maturity” by 2024 and for the period up to 2030]. Available at: https://www.tadviser.ru/images/8/83/Strateg_info_2021_compressed.pdf (accessed 07 November 2024).
2. Korchevskaya A.A. [i dr.]. Trendy i stsennariy razvitiya rynka sistem rasshirennoy planirovaniya proizvodstva (APS-sistem) v ramkakh napravleniya «Tekhnety» NTI. Ekspertno-analiticheskiy dok-lad: monografiya [Trends and scenarios for the development of the market of advanced production planning systems (APS systems) within the framework of the Technet direction of the NTI. Expert and analytical report: monograph]. Saint Petersburg: POLITEKH-PRESS, 2024, 112 p.
3. Obukhov K.O. Sravnitel'nyy analiz sistem proizvodstvennogo planirovaniya v razreze raboty s al'ternativnymi tsepochkami proizvodstvennykh operatsiy [Comparative analysis of production planning systems in the context of working with alternative chains of production operations], *Molodoy uchenyy* [Young scientist], 2022, No. 20 (415), pp. 143-147. Available at: <https://moluch.ru/archive/415/91773/> (accessed 07 November 2024).
4. Klimova E.Z., Pavlova I.A., Makusheva Yu.A. Mesto i rol' operativno-proizvodstvennogo planirovaniya v upravlenii predpriyatiem [Place and role of operational production planning in enterprise management], *EGI* [EGI], 2022, No. 42 (4). Available at: <https://cyberleninka.ru/article/n/mesto-i-rol-operativno-proizvodstvennogo-planirovaniya-v-upravlenii-predpriyatiem> (accessed 07 November 2024).
5. Dzhukha V.M., Kirillov D.O. Potentsial'nye preimushchestva APS-sistem v prodazhakh i operatsionnom planirovanii [Potential advantages of APS systems in sales and operational planning], *UEkS* [Management of economic systems], 2013, No. 7 (55). Available at: <https://cyberleninka.ru/article/n/potentsialnye-preimushchestva-aps-sistem-v-prodazhah-i-operatsionnom-planirovanii-1> (accessed 10 December 2024).
6. Lazarev A.A., Gafarov E.R. Teoriya raspisaniy. Zadachi i algoritmy [Scheduling theory. Problems and algorithms]. Moscow: Fizicheskii fakul'tet MGU, 2011, 222 p.
7. Job shop scheduling: Classification, constraints and objective functions. Available at: https://www.researchgate.net/publication/339128833_Job_shop_scheduling_Classification_constraints_and_objective_functions (accessed 07 November 2024).
8. A review on job shop scheduling with setup times. Available at: https://www.researchgate.net/publication/275317984_A_review_on_job_shop_scheduling_with_setup_times (accessed 10 December 2024).
9. Planirovanie i optimizatsiya protsessov perenaladki [Planning and optimization of changeover processes]. Available at: <https://habr.com/ru/companies/ds/articles/533788> (accessed 10 December 2024).
10. A Flexible Branch and Bound Method for the Job Shop Scheduling Problem. Available at: https://www.researchgate.net/publication/238723625_A_Flexible_Branch_and_Bound_Method_for_the_Job_Shop_Scheduling_Problem (accessed 07 November 2024).
11. Dynamic Programming for Routing and Scheduling. Available at: <https://research.vu.nl/ws/portalfiles/portal/42163578/complete+dissertation.pdf> (accessed 07 November 2024).
12. Designing Dispatching Rules to Minimize Total Tardiness. Available at: https://www.researchgate.net/publication/226253624_Designing_Dispatching_Rules_to_Minimize_Total_Tardiness (accessed 07 November 2024).
13. A Job-Shop Scheduling Problem (JSSP) using Genetic Algorithm (GA). Available at: https://www.researchgate.net/publication/242081421_A_JOB-SHOP_SCHEDULING_PROBLEM_JSSP_USING_GENETIC_ALGORITHM_GA (accessed 07 November 2024).
14. Sochnev A.N. Planirovanie proizvodstva na osnove upravlyayemoy geneticheskim algoritmom imitatsionnoy modeli [Production planning based on a simulation model controlled by a genetic algorithm], *Zhurnal SFU. Tekhnika i tekhnologii* [Journal of SFU. Engineering and Technology], 2021, No. 2. Available at: <https://cyberleninka.ru/article/n/planirovanie-proizvodstva-na-osnove-upravlyayemoy-geneticheskim-algoritmom-imitatsionnoy-modeli> (accessed 07 November 2024).

15. Solving the Job Shop Scheduling Problem with Ant Colony Optimization. Available at: https://www.researchgate.net/publication/363500511_Solving_the_Job_Shop_Scheduling_Problem_with_Ant_Colony_Optimization (accessed 07 November 2024).
16. Zhang F., Mei Y., Nguyen S. and Zhang M. Survey on Genetic Programming and Machine Learning Techniques for Heuristic Design in Job Shop Scheduling, in *IEEE Transactions on Evolutionary Computation*, Feb. 2024, Vol. 28, No. 1, pp. 147-167.
17. Dynamic flexible job shop scheduling method based on improved gene expression programming. Available at: https://www.researchgate.net/publication/343612768_Dynamic_flexible_job_shop_scheduling_method_based_on_improved_gene_expression_programming (accessed 10 December 2024).
18. Zhang F., Mei Y., Nguyen S. and Zhang M. Evolving Scheduling Heuristics via Genetic Programming With Feature Selection in Dynamic Flexible Job-Shop Scheduling, in *IEEE Transactions on Cybernetics*, April 2021, Vol. 51, No. 4, pp. 1797-1811.
19. Genetic Programming: An Introduction and Tutorial. Available at: https://wiki.eecs.yorku.ca/course_archive/2010-11/W/4403/_media/gp1.pdf (accessed 07 November 2024).
20. Polikarpova N.I., Tochilin V.N. Primenenie geneticheskogo programmirovaniya dlya realizatsii sistem so slozhnym povedeniem [Application of genetic programming for the implementation of systems with complex behavior], *Nauchno-tekhnicheskiiy vestnik informatsionnykh tekhnologiy, mekhaniki i optiki* [Scientific and technical bulletin of information technologies, mechanics and optics], 2007, No. 39. Available at: <https://cyberleninka.ru/article/n/primenenie-geneticheskogo-programmirovaniya-dlya-realizatsii-sistem-so-slozhnym-povedeniem> (accessed 07 November 2024).

Обухов Константин Олегович – Астраханский государственный технический университет; e-mail: kobukhov98@mail.ru; г. Астрахань, Россия; кафедра автоматизированных систем обработки информации и управления; аспирант.

Квятковская Ирина Юрьевна – e-mail: i.kvyatkovskaya@astu.org; кафедра высшей и прикладной математики; д.т.н.; профессор.

Морозов Александр Васильевич – e-mail: morozov@ilabsltd.com; кафедра автоматизированных систем обработки информации и управления; к.т.н.; доцент.

Obukhov Konstantin Olegovich – Astrakhan State Technical University; e-mail: kobukhov98@mail.ru; Astrakhan, Russia; the Department of Automated Information Processing and Control Systems; postgraduate student.

Kvyatkovskaya Irina Yurievna – e-mail: i.kvyatkovskaya@astu.org; the Department of Higher and Applied Mathematics; dr. of eng. sc.; professor.

Morozov Aleksandr Vasil'evich – e-mail: morozov@ilabsltd.com; the Department of Automated Information Processing and Control Systems; cand. of eng. sc.; associate professor.

УДК 621.37

DOI 10.18522/2311-3103-2024-6-52-64

З.А. Понимаш, М.В. Потанин

МЕТОД И АЛГОРИТМ ИЗВЛЕЧЕНИЯ ПРИЗНАКОВ ИЗ ЦИФРОВЫХ СИГНАЛОВ НА БАЗЕ НЕЙРОСЕТЕЙ ТРАНСФОРМЕР

В последнее время нейросетевые модели стали одним из наиболее перспективных направлений в области автоматического извлечения признаков из цифровых сигналов. Традиционные подходы, такие как статистический, временной, частотный и частотно-временной анализ, требуют значительных экспертных знаний и часто оказываются недостаточно эффективными при работе с нестационарными и сложными сигналами, например, биомедицинскими (ЭКГ, ЭЭГ, ЭМГ) или промышленными сигналами (примером могут служить токограммы). Перечисленные выше методы имеют ряд ограничений, когда требуется анализировать многоканальные данные с изменяющейся частотной структурой, либо когда разметка сигналов слишком трудоёмка или дорогостоящая. Современные архитектуры нейросетей, такие как трансформеры, показали высокую эффективность в автоматическом извлечении признаков из сложных данных. Трансформеры превосходят традиционные свёрточные и рекуррентные нейронные сети по многим ключевым характеристи-