

УДК 519.8, 004.9

А.Г. Гривачевский¹, В.Д. Гривачевский², М.Я. Ковалев¹, А.И. Сухомлинов²

ПЛАНИРОВАНИЕ ПРОЦЕССА ИЗГОТОВЛЕНИЯ СЕКЦИИ КОРПУСА СУДНА

Рассматривается задача планирования процесса изготовления секции корпуса судна, которая сводится к классической в теории расписаний трехстадийной задаче Джонсона. Описывается генетический алгоритм решения и компьютерный эксперимент для сравнения эффективности этого алгоритма и алгоритма полного перебора.

Введение

Составление производственных планов и расписаний является одной из актуальных задач управления деятельностью судостроительного предприятия. В статье рассматривается процесс изготовления деталей секции корпуса судна и предлагается сведение задачи оптимального планирования этого процесса к классической задаче теории расписаний для трех последовательных станков, обозначаемой $F3//C_{\max}$, – трехстадийной задаче Джонсона [1]. Поскольку задача является NP -трудной в сильном смысле [2], для ее решения предлагается генетический алгоритм.

1. Практическая ситуация и трехстадийная задача Джонсона

В основу современной организации производства сварного корпуса судна положен принцип декомпозиции конструкции корпуса на секции и независимого параллельного изготовления этих секций с последующей сборкой их в общий корпус. Рассмотрим задачу планирования процесса изготовления одной секции корпуса судна. Этот процесс включает изготовление необходимых для секции узлов из листового металлопроката и их сварку. Каждый узел состоит из определенного набора деталей. Все детали при изготовлении проходят три производственные стадии:

1. Разметка детали на металлическом прокате.
2. Вырезка детали.
3. Обработка детали после вырезки.

Детали проходят указанные три стадии в одной и той же последовательности: 1, 2, 3. На одной и той же стадии производственные операции над различными деталями выполняются последовательно и не могут пересекаться по времени. Кроме того, операции различных стадий над одной и той же деталью не могут пересекаться во времени, однако операции над различными деталями могут выполняться параллельно во времени на различных стадиях. Предположим, что количество деталей равно n , они обозначаются $j = 1, \dots, n$, и для каждой детали j известны длительности p_{ij} выполнения соответствующей операции на стадии i , $i = 1, 2, 3$.

В табл. 1 приведен пример значений длительностей операций при изготовлении 10 деталей секции корпуса судна на Дальневосточном заводе «Звезда». Объектом планирования является расписание выполнения операций над n деталями на указанных трех стадиях, которое, в свою очередь, определяется указанием моментов времени C_{ij} завершения выполнения операций над деталями j на стадиях i . Пусть C_j обозначает момент завершения последней операции над деталью j , т. е. $C_j = C_{3j}$. Исходя из предположения, что выполнение любой операции может начаться в момент времени ноль, задача заключается в поиске расписания, минимизирующего момент времени C_{\max} завершения выполнения последней операции над всеми деталями, $C_{\max} = \max\{C_{3j}/j=1, \dots, n\}$, называемый также длиной расписания.

Таблица 1

Длительности выполнения операций, мин

Номер детали	Разметка	Вырезка	Обработка
1	4	5	8
2	3	6	7
3	9	1	4
4	6	9	1
5	4	1	6
6	7	8	9
7	1	7	6
8	8	1	8
9	3	1	3
10	4	9	1

Сформулированная задача как классическая задача теории расписаний $F3//C_{\max}$ была впервые поставлена Джонсоном [1], который доказал, что оптимальное расписание находится в классе так называемых перестановочных расписаний, для которых последовательность операций на каждой стадии одинакова и определяется последовательностью деталей.

2. Генетический алгоритм

Для решения задачи $F3//C_{\max}$ предлагается воспользоваться генетическим алгоритмом, суть которого заключается в генерации случайным образом множества (начальной популяции) последовательностей деталей (хромосом) и получении решения (искомой оптимальной последовательности деталей) при помощи генетических операторов скрещивания и мутации. Генетические алгоритмы не гарантируют построение оптимального решения, однако в большинстве случаев позволяют отыскать решение, близкое к оптимальному. Основы генетических алгоритмов описаны Холландом [3]. Анализ литературы показывает эффективность и целесообразность применения данного метода для решения задач планирования работ современных судостроительных предприятий [4, 5].

Хромосомой разрабатываемого генетического алгоритма будем считать последовательность деталей (например, 2, 3, 7, 6, 4, 1, 10, 5, 9, 8).

В качестве фитнес-функции используется целевая функция C_{\max} , которая для любой последовательности деталей может быть вычислена рекурсивно следующим образом. Не ограничивая общности, рассмотрим последовательность $(1, \dots, n)$. Введем вспомогательные величины fin_1 , fin_2 и fin_3 . Вначале полагаем $fin_1 = 0$, $fin_2 = 0$, $fin_3 = 0$. Далее для $j = 1, \dots, n$ рекуррентно вычисляем

$$\begin{aligned} fin_1 &:= fin_1 + p_{1j}, \\ fin_2 &:= fin_2 + p_{2j} + \max\{fin_1 - fin_2, 0\}, \\ fin_3 &:= fin_3 + p_{3j} + \max\{fin_2 - fin_3, 0\}. \end{aligned}$$

Значение фитнес-функции $C_{\max} = fin_3$ при $j = n$.

Скрещивание и мутация осуществляются только для одной и той же хромосомы, так как могут возникнуть ситуации, когда при скрещивании двух различных хромосом или мутации полученная хромосома (последовательность деталей) будет содержать повторяющиеся номера деталей.

При скрещивании для определенной хромосомы выбирается точка между номерами соседних генов (позиций в перестановке деталей), относительно которой происходит обмен местами двух частей родительской хромосомы. Допустим, для хромосомы, представленной на рис. 1, выбирается точка скрещивания между четвертым и пятым генами, в результате скрещивания получаем новую хромосому. При этом в новую популяцию попадают обе хромосомы – «родитель» и «потомок».

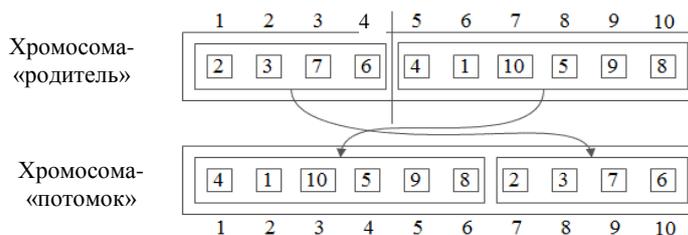


Рис. 1. Скрещивание хромосом в разработанном методе

При мутации для определенной хромосомы выбираются номера генов, которые затем производят обмен значениями между собой. Например, для хромосомы, представленной на рис. 2, выбираются номера генов 3 и 8 для мутации, в результате чего получаем новую хромосому. При этом в новую популяцию попадают обе хромосомы – до и после мутации.

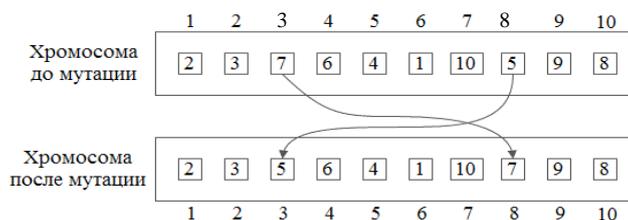


Рис. 2. Мутация хромосом в разработанном методе

Для исходных данных в табл. 1 с помощью описанных операций скрещивания и мутации хромосом были получены пять различных расписаний одинаковой длины $C_{\max} = 59$ мин (рис. 3). В целях практического улучшения получаемых расписаний был введен дополнительный критерий Idle (простой), значением которого является суммарное время промежуточного простоя в минутах для всех трех стадий между моментами начала и завершения работ для каждой стадии. На рис. 3 промежуточные простои обозначаются символом 0.

Заметим, что для любого расписания операции первой стадии могут быть смещены по оси времени максимально влево и операции третьей стадии – максимально вправо так, что расписание останется допустимым, значение C_{\max} не увеличится, а промежуточные простои на первой и третьей стадиях исчезнут. Все построенные алгоритмом расписания трансформируем указанным образом (рис. 4). Для расписания 4, а значение критерия Idle равно 6 мин, для 4, б – 4 мин, для 4, в – 6 мин, для 4, г – 6 мин, для 4, д – 0 мин. Среди расписаний с одинаковой длиной предпочтительным является расписание с наименьшим промежуточным простоем.

Предлагается рассмотреть двухмерную фитнес-функцию (C_{\max} , Idle) с лексикографическим отношением предпочтения, таким, что значение C_{\max} является первичным (более значимым), а значение Idle – вторичным. Функцию Idle необходимо минимизировать на множестве расписаний, минимизирующих C_{\max} .

Для определения экономической выгоды использования разработанного метода был проведен сравнительный анализ наихудшего расписания, соответствующего последовательности 9, 3, 8, 4, 7, 10, 6, 1, 2, 5, и расписания на рис. 4, д, полученного с помощью генетического алгоритма. Сравнительный анализ этих расписаний приведен в табл. 2.

Таблица 2

Сравнение полученного и наихудшего расписаний

Параметр	Наилучший план	Наихудший план
Время окончания работ по разметке деталей, мин	49	49
Время окончания работ по вырезке деталей, мин	52	71
Время окончания работ по обработке деталей, мин	59	89
Итоговое время, мин	59	89
Промежуточный простой при вырезке деталей, мин	0	20

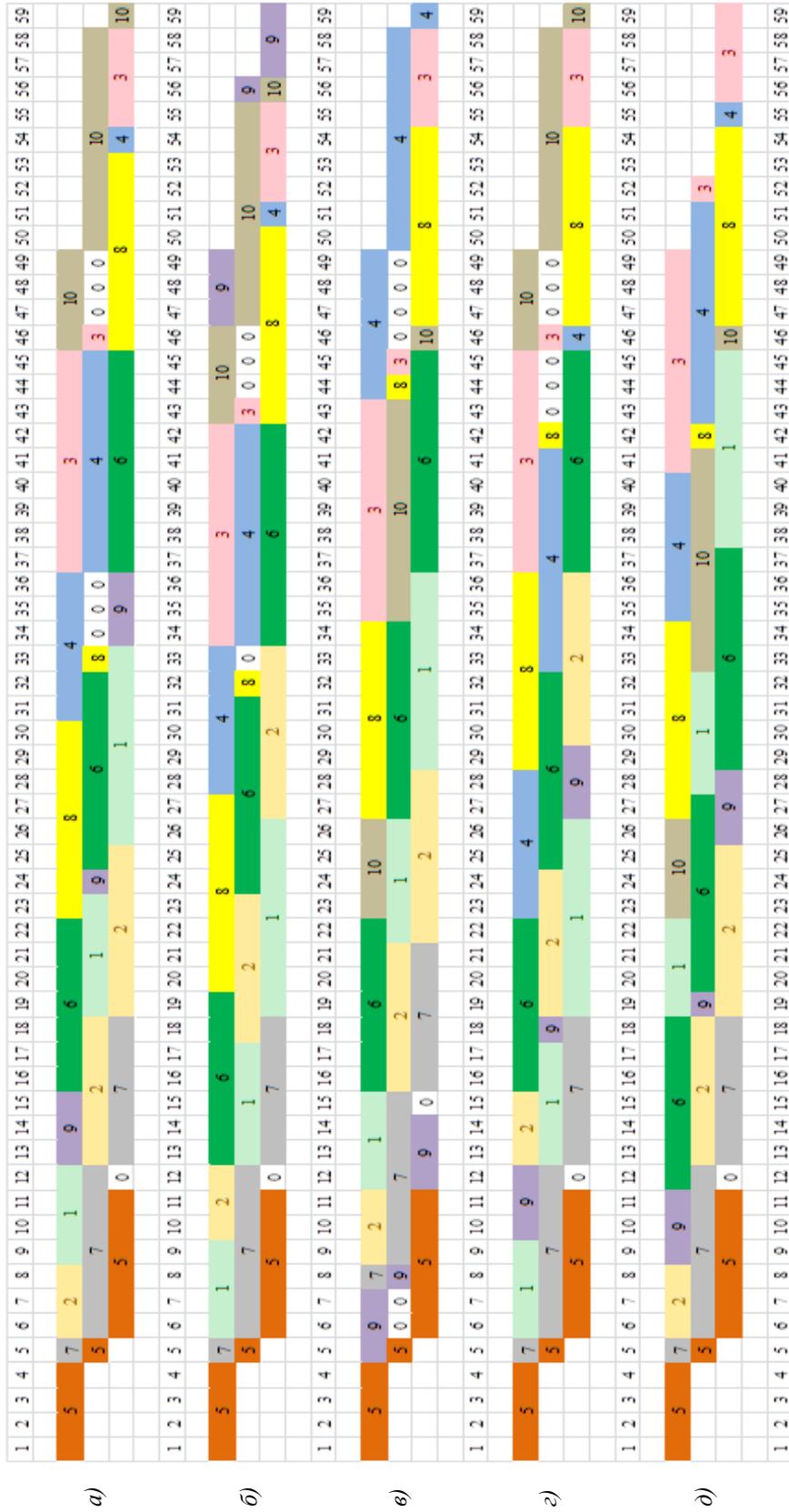


Рис. 3. Диаграммы Ганта для расписаний производственного процесса, полученных с помощью генетического алгоритма

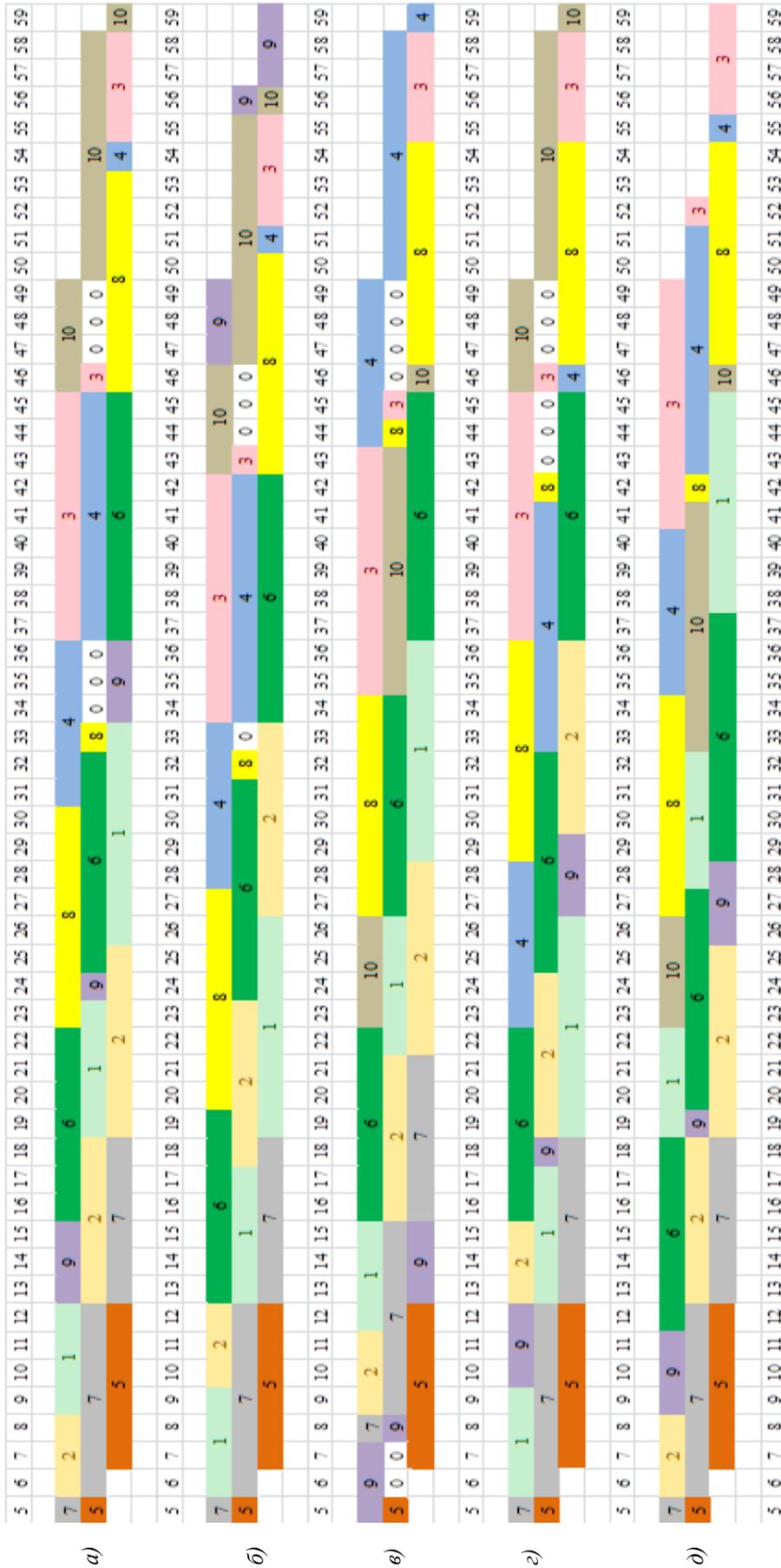


Рис. 4. Диаграммы Ганта для расписаний производственного процесса, полученных с помощью генетического алгоритма, со смещением времени начала работ на первой и третьей стадиях

Исходя из полученных результатов, можно сделать вывод, что по критерию итогового времени наилучшее расписание обеспечивает изготовление всех деталей быстрее на 33,7 %, снижая промежуточный простой с 20 мин до нуля (на 100 %).

Генетический алгоритм и алгоритм полного перебора были разработаны в среде Microsoft Visual Studio 2012 на языках C# и C++ соответственно. Параметры генетического алгоритма: размер популяции – 1000 особей, количество итераций – 1000.

3. Вычислительные эксперименты

Вычислительные эксперименты проводились на персональном компьютере с операционной системой Windows 7 Ultimate SP1 32-bit, центральным процессором Intel Core 2 DUO E8400 3,00 ГГц и оперативной памятью 4 Гб. Было проведено исследование скорости нахождения решения для исходных данных из табл. 1 с помощью генетического алгоритма и алгоритма полного перебора. При заданном количестве деталей n входом являлись детали 1, ..., n . Сравнительные результаты экспериментов представлены в табл. 3. Отметим, что во всех случаях генетический алгоритм нашел оптимальное решение.

Таблица 3

Результаты экспериментов

Количество деталей	Количество перестановок	Генетический алгоритм		Полный перебор	
		Время работы, с	C_{\max} , Idle	Время работы, с	C_{\max} , Idle
4	$4! = 24$	5	29,0	< 1	29,0
5	$5! = 120$	6	33,4	< 1	33,4
6	$6! = 720$	7	42,4	1	42,4
7	$7! = 5040$	8	47,1	1	47,1
8	$8! = 40\,320$	9	55,2	3	55,2
9	$9! = 362\,880$	10	58,4	26	58,4
10	$10! = 3\,628\,800$	11	59,0	260	59,0

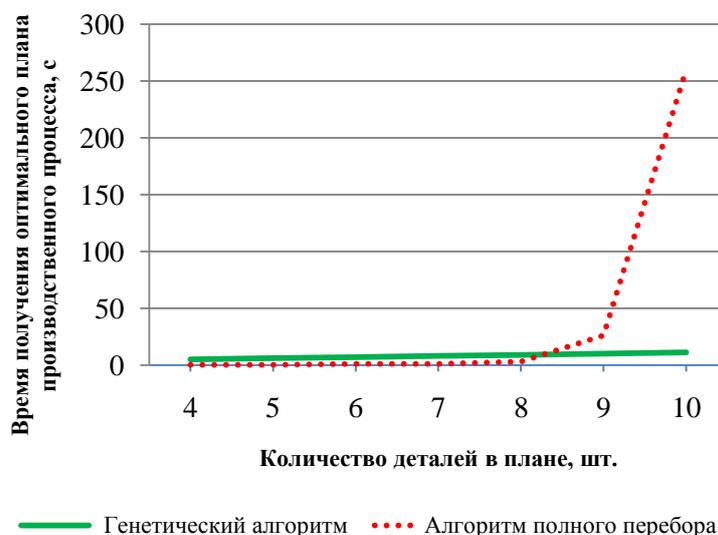


Рис. 5. График зависимости времени получения решения от количества деталей

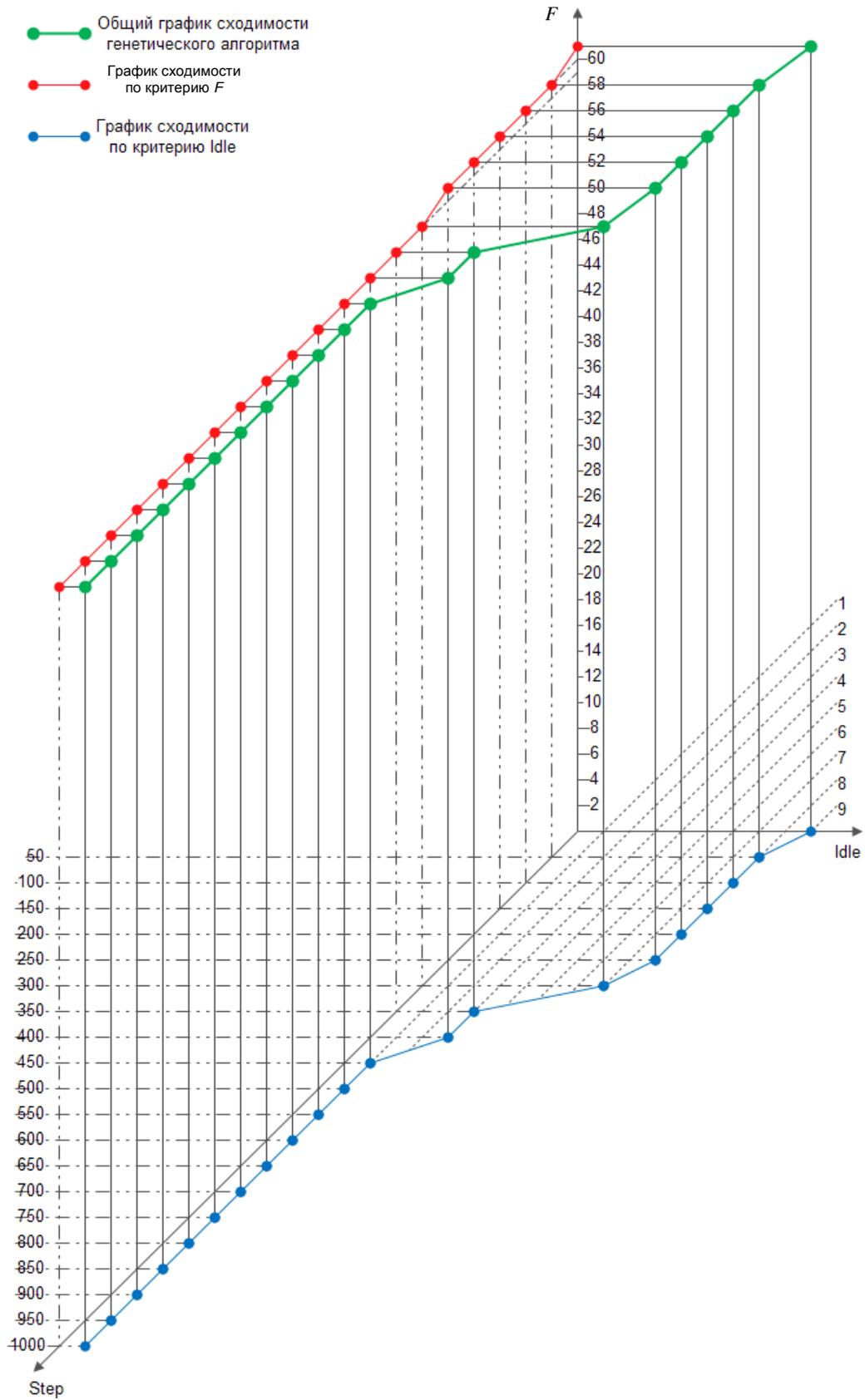


Рис. 6. График сходимости генетического алгоритма

Из рис. 5 видно, что на полный перебор при количестве деталей от 4 до 8 затрачивается в разы меньше времени, чем на реализацию генетического алгоритма. Однако, учитывая, что масштабы производства достаточно велики, при большом количестве деталей генетический алгоритм сохраняет практически линейную зависимость времени получения решения от количества деталей, тогда как время, необходимое для полного перебора, увеличивается на порядок с каждой дополнительной деталью.

На рис. 6 представлен график сходимости генетического алгоритма по критериям C_{\max} (ось F) и $Idle$ в зависимости от номера шага выполнения алгоритма (ось $Step$) для примера на рис. 4, d . Результирующая кривая есть функция от двух переменных. Для удобства восприятия сходимости по каждому из критериев на рис. 6 представлены также проекции общего графика сходимости на плоскость F - $Step$ для критерия C_{\max} и на плоскость $Idle$ - $Step$ для критерия $Idle$. Видно, что решение было получено на 401-м – 450-м шагах выполнения алгоритма.

Заключение

Применение современных методов автоматизированного планирования производственного процесса, основанных на генетических алгоритмах, позволяет получить оптимальные и субоптимальные планы, графики и расписания деятельности предприятия за приемлемое время, что приводит к сокращению затрат временных и материальных ресурсов, тем самым увеличивая эффективность производства и производительность труда инженеров-технологов.

Список литературы

1. Johnson, S.M. Optimal two- and three-stage production schedules with set-up time included / S.M. Johnson // *Naval Research Logistics Quarterly*. – 1954. – Vol. 1. – P. 61–68.
2. Гэри, М. Вычислительные машины и труднорешаемые задачи / М. Гэри, Д. Джонсон. – М. : Наука, 1979.
3. Holland, J. *Adaptation in natural and artificial systems* / J. Holland. – Cambridge : MIT Press, 1992.
4. Xu, X. Computer-aided process planning – A critical review of recent developments and future trends / X. Xu, L. Wang, S.T. Newman // *International Journal of Computer Integrated Manufacturing*. – 2011. – Vol. 24, № 1. – P. 1–31.
5. Qassim, R.Y. Integrated process planning and scheduling and multimode resource constrained project scheduling: ship block assembly application / R.Y. Qassim // *Manufacturing System*. – 2012. – P. 307–314.

Поступила 10.09.2015

¹Объединенный институт проблем информатики НАН Беларуси,
Минск, Сурганова, 6
e-mail: griva@newman.bas-net.by

²Дальневосточный федеральный университет,
Владивосток, ул. Суханова, 8

A.G. Grivachevsky, V.D. Grivachevsky, M.Y. Kovalyov, A.I. Suhomlinov

PLANNING VESSEL BODY SECTION PRODUCTION

A problem of planning production of a vessel body section is considered. The problem is reduced to the classic Johnson's tree-machine flow-shop scheduling problem. A genetic algorithm and computer experiment to compare efficiency of this algorithm and the algorithm of full enumeration are described.