

УДК 51.330.115

ИМИТАЦИОННАЯ МОДЕЛЬ ДЛЯ РЕШЕНИЯ ЗАДАЧ  
КАЛЕНДАРНОГО ПЛАНИРОВАНИЯ МАШИНОСТРОИТЕЛЬНОГО  
ПРОИЗВОДСТВА

Ю. М. Зыбарев

На многих машиностроительных предприятиях остро стоит проблема дефицита рабочих. На производственных участках численность рабочих, как правило, значительно меньше числа единиц оборудования. Закрепление за каждым рабочим по одному станку здесь невозможно, поэтому рабочие вынуждены обслуживать различные станки. В этих условиях сложность процессов принятия решений по управлению производством на участках значительно возрастает, так как задачи, решаемые при этом управляющим органом, приобретают более сложный комбинаторный характер. В отличие от задач "операция-станок", которые обычно рассматриваются в многочисленных работах по календарному планированию (теории расписаний) [1 - 4], для предлагаемого класса задач характерно то, что кроме согласования сроков запуска в обработку по операциям на единицах оборудования в соответствии с технологией необходимо согласование периодов занятости рабочих. Ясно, что обычная задача календарного планирования "операция-станок" является частным случаем, когда каждый станок обслуживается от-

дельным рабочим.

В предлагаемой работе рассматривается метод имитационного моделирования в приложении к исследованию процессов решения задач календарного планирования машиностроительного производства, когда численность рабочих на участках значительно меньше числа единиц оборудования. При этом предполагается, что элементы производства не подвергаются воздействиям случайного характера в процессе функционирования, т.е. характеризуются вполне детерминированным "поведением".

В § 1,2 излагается описание имитационной модели, в § 3 показывается на примерах, каким образом эта модель может быть использована в исследовании различных аспектов решения задач календарного планирования.

## § I. Описание объекта управления

Объектом управления на участке машиностроительного предприятия является производство, которое будем описывать классом производственных процессов  $W$  и двумя классами ресурсов:

$i \in I = \{1, 2, \dots, i^*\}$  — идентификаторы рабочих;  $j \in J = \{1, 2, \dots, j^*\}$  — идентификаторы единиц оборудования.

Класс производственных процессов  $W$  представляет собой частично упорядоченное множество состояний предметов труда, которые они принимают на пути от исходных заготовок (материалов) до готового продукта. Различаются два вида состояний:

а) состояния обработки предмета труда, которые характеризуются множеством операций обработки  $R$ ;

б) переходные состояния (состояния межоперационного пролеживания), которые характеризуются элементами множества  $V$ .

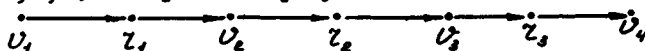
Каждому из состояний предметов труда  $(RUV)$  ставится в соответствие индекс заказа  $d$ ,  $d = 1, 2, \dots, d^*$ , который определяет конструктивные единицы (детали, узлы и т.д.) изделий, выпускаемых предприятием. Согласно индексу заказа класс производственных процессов  $W$  распадается на совокупность частичных процессов  $\{W_d | d = 1, 2, \dots, d^*\}$ , где  $W_d$  определяется подмножествами  $R_d \subseteq R$ ,  $V_d \subseteq V$  состояний с индексом заказа  $d$ .

Если ограничить круг рассматриваемых производственных участков только теми, у которых технологические процессы однозначно определяют порядок выполнения операций обработки предме-

тов труда, то вполне естественно описывать структуру производственных процессов  $W$  ориентированным, вообще говоря, несвязным графом  $\Gamma_W = (R, V, U)$ . Подмножество вершин графа определяется подмножествами состояний предметов труда  $R$  и  $V$ , а дуги  $U$  графа описывают порядок перехода предмета труда из одного состояния в другое, которые предопределяются конструкторско-технологической и организационной документацией.

Поясним на примере. Участок производит три наименования деталей, для производства которых требуется выполнять соответственно 3, 2 и 4 операции механообработки. Множество заказов для этого участка состоит из трех элементов  $d=1, 2, 3$ , каждый из которых соответствует отдельному наименованию детали. Множество операций состоит из девяти элементов  $R=\{z_1, z_2, \dots, z_9\}$ , где  $\{z_1, z_2, z_3\} = R_{d=1}$ ,  $\{z_4, z_5\} = R_{d=2}$  и  $\{z_6, z_7, z_8, z_9\} = R_{d=3}$  - подмножества операций с соответствующими индексами заказа. Порядок выполнения операций в пределах каждого заказа, задаваемый технологией, совпадает с последовательностями  $R_d$ ,  $d=1, 2, 3$ . Так как сборочные операции на участке отсутствуют, то подмножества переходных состояний предметов труда (деталей) определяются через  $V_{d=1}=\{v_1, v_2, v_3, v_4\}$ ,  $V_{d=2}=\{v_5, v_6, v_7\}$  и  $V_{d=3}=\{v_8, v_9, v_{10}, v_{11}, v_{12}\}$ . Состояние детали  $d=1$ , которое характеризует поступившие на операцию  $z_1$  заготовки, идентифицируется переходным состоянием  $v_1 \in V_{d=1}$ , а промежуточные между операциями  $z_1$  и  $z_2$ ,  $z_2$  и  $z_3$  - состояниями  $v_2, v_3 \in V_{d=1}$ . Готовой детали  $d=1$ , прошедшей обработку на всех операциях, соответствует  $v_4 \in V_{d=1}$ .

Аналогично определяются элементы подмножеств  $V_{d=2}$ ,  $V_{d=3}$ . Таким образом, совокупность производственных процессов  $W$  на участке состоит из трех частичных процессов  $W_d = \{R_d, V_d\}$ ,  $d=1, 2, 3$ . Частичный процесс  $W_{d=1}$  в виде фрагмента графа  $\Gamma_W = (R, V, U)$  изображен на рисунке



Полный граф  $\Gamma_W$  для данного участка состоит из трех аналогичных, не связанных между собой фрагментов, соответствующих частичным процессам  $W_d$ ,  $d=1, 2, 3$ .

Операция обработки  $z \in R$  кроме индекса заказа  $d$  характеризуется следующими параметрами:

$t_z^H$  - длительностью реализации единичного акта обработки;

$I_z = \{i / i \in I\} \subseteq I$  - подмножеством номеров рабочих, которые могут выполнить операцию  $z$ ;

$J_z \subseteq J$  - подмножеством единиц оборудования, на которых может быть выполнена операция  $z$ ;

$V_z^-$  - подмножеством переходных состояний предметов труда, из которых заготовки поступают на операцию  $z$ ;

$V_z^+$  - переходным состоянием предметов труда, в которое поступают они после обработки на операции  $z$ ;

$r_z^V = \{r_z^v / v \in V_z^-\}$  - набором чисел, описывающих норму расхода предметов труда с переходных состояний  $v \in V_z^-$  на единичный акт реализации операции  $z$  (при этом считаем, что в результате реализации единичного акта обработки получается единица продукта);

$C_z''$  - расценкой на реализацию единичного акта операции обработки  $z$ ;

$\delta_z$  - признаком вида операции, который определяет характер обработки:штучный ( $\delta_z = 0$ ) или партионный ( $\delta_z = 1$ ); примером партионной обработки служат транспортная операция, термообработка и т.д.

Характерной чертой машиностроительного производства является повторяемость выполнения заказа  $d$  в процессе производства в течение некоторого периода времени, а также реализация заказа некоторой расчетной партией. Для их идентификации введем понятие детали-партии  $(d, k)$ , которое и будет характеризовать отдельную реализацию заказа  $d, k = 1, 2, \dots, n_d$  в объеме  $y_{d,k}$ , где  $d \in D^0$ ;  $D^0 = \{1, 2, \dots, d^*\}$ , - директивно заданная к обработке и выпуску номенклатура заказов в рассматриваемый период времени.

Реализация операций обработки  $z \in R$  связана с понятием технологического способа производства (ТСП)  $g = (d, k, z, i, j)$ , где элементы ТСП определяют, каким рабочим  $i \in I_z$  на какой единице оборудования  $j \in J_z$  выполняется операция  $z$  детали-партии  $(d, k)$ . ТСП характеризуется параметрами:

$t_g^s$  - временем, начиная с которого осуществляется процесс реализации ТСП  $g$ ;

$t_g^e$  - временем, до которого осуществляется процесс реализации ТСП  $g$ .

Интенсивность  $y_g$  реализации ТСП  $g$  определяется объемом

обрабатываемой детали-партии  $y_{d,k}$ . Если обозначить через  $T_e$  текущий момент времени, то способ производства  $G^*(T_e) = \{g | t_g^* \leq T_e, t_g^* > T_e\}$  будут называться активными.

Состояния предметов труда  $v \in V$  кроме параметров, связывающих это понятие с операциями обработки  $z \in K$ , которые определены выше, описываются параметром  $\alpha_v(T_e)$ , определяющим наличие предметов труда в переходном состоянии  $v$  на момент времени  $T_e$ .

Рабочий  $i \in I$  описывается индивидуальной производительностью труда  $\eta_i$  и параметром текущего состояния занятости  $\alpha_i(T_e)$ , где  $0 \leq \alpha_i(T_e) \leq 1$ . Причем  $\alpha_i(T_e) = 0$  означает, что в момент  $T_e$  рабочий  $i$  полностью занят в активных ТСП, а  $0 < \alpha_i(T_e) \leq 1$  определяет полную или частичную незанятость рабочего в момент  $T_e$ . Кроме того, рабочий  $i$  имеет дополнительные характеристики текущего состояния, которые включают:

$\tau_i^{np}(T_e)$  - длительность суммарных простоев рабочего  $i$  с начала планового периода на момент  $T_e$ ;

$c_i^6(T_e)$  - суммарную выработку в рублях рабочего с начала планового периода на момент  $T_e$ .

Единица оборудования  $j \in J$  описывается следующим набором параметров:

$t_j^{нал}$  - длительностью переналадки единицы оборудования с одной операции на другую;

$\beta_j$  - степень занятости рабочего при обслуживании им данной единицы оборудования  $j$ ,  $0 < \beta_j \leq 1$ , где значение  $\beta_j = 1$  определяет одностаночный режим обслуживания;

$\beta_j^0(T_e)$  - текущим состоянием занятости единицы оборудования на момент  $T_e$  в активных ТСП, где  $0 \leq \beta_j^0(T_e) \leq \beta_j^*$ ,  $\beta_j^*$  - максимальное число операций, которые могут одновременно выполняться на  $j$ ;

$\tau_j^{np}(T_e)$  - суммарной длительностью простоев  $j$ -й единицы оборудования с начала планового периода на текущий момент  $T_e$ .

Описывая текущее состояние элементов производства, определим параметры, характеризующие введенные выше детали-партии  $(d, k)$ :

$t_{d,k}(T_e)$  - признак вида движения детали-партии в процессе обработки, где предусматривается два вида движения: последова-

тельный  $f_{d,k}(T_e)=1$  и параллельный  $f_{d,k}(T_e)=0$ ;  
 $z_{d,k}^0(T_e)$  - номер первой невыполняемой на момент  $T_e$  операции  $z \in R_{d,k}$  детали-партии  $(d, k)$ ;

$t_{d,k,z}^e$  - время готовности к началу обработки детали-партии на операции  $z = z_{d,k}^0(T_e)$  (для первой операции параметр определяет срок поступления заготовок по детали-партии на участок).

В зависимости от признака вида движения детали-партии  $f_{d,k}(T_e)$  значение параметра  $t_{d,k,z}^e$  для операции  $z = z_{d,k}^0(T_e)$  определяется следующими выражениями:

а) для последовательного вида движения

$$t_{d,k,z}^e = t_g^e + t_{z-1}^H \cdot [y_{d,k} - x_o(t_g^e)], \quad (I.1)$$

где  $g = (d, k, (z-1), i, j)$ ,  $v \in V_z^-$ ;

б) для параллельного вида движения

$$t_{d,k,z}^e = t_g^e + \max\{0; [\max\{t_{z-1}^H; (y_{d,k} - x_o(t_g^e)) \cdot t_{z-1}^H - \\ - (y_{d,k} - 1) \cdot t_{z-1}^H\} - x_o(t_g^e) \cdot t_{z-1}^H]\}, \quad (I.2)$$

где  $g = (d, k, (z-1), i, j)$ ,  $v \in V_z^-$ .

Таким образом, каждому моменту времени  $T_e$  соответствует некоторое состояние производства  $S(T_e) = \{S_i(T_e), S_j(T_e), S_{d,k}(T_e), x_o(T_e), G^*(T_e) | i \in I, j \in J, d \in D^0, k = 1, 2, \dots, n_d\}$ , где компоненты характеризуют текущее состояние элементов производства. Последовательность  $\{S(T_0), S(T_1), \dots, S(T_N)\}$  описывает процесс функционирования производства на временном интервале  $[T_0, T_N]$ . Состояние  $S(T_0)$  определяет начальное состояние элементов производства перед процессом моделирования.

## § 2. Алгоритм имитационной модели

Процесс производства на участке был нами представлен как упорядоченная последовательность состояний  $\{S(T_0), S(T_1), \dots, S(T_N)\}$ ;  $T_e \in T = \{T_0, T_1, \dots, T_N\}$ . Из предположения об отсутствии влияния на элементы производства случайных воздействий следует, что процесс функционирования производства полностью предопределяется результатом деятельности управляющего органа на участке, т.е. последующие состояния производства определяются совокупностью принятых управляющим органом решений в текущий и все предшествующие ему моменты времени. Поэтому в

предлагаемой имитационной модели модельное время  $T_e \in T$  связывается с наличием на участке производственной ситуации. в которой необходимо принять решения.

Производственной ситуацией на участке будем считать такое состояние производства  $S(T_e)$ , для которого выполнены следующие условия:

- 1)  $I_o(T_e) \neq \emptyset$ , где  $I_o(T_e) = \{i | d_i(T_e) > 0\}$  - подмножество рабочих, которые в момент  $T_e$  не заняты (полностью или частично) в активных ТСП  $G^*(T_e)$ ;
- 2)  $J_o(T_e) = \emptyset$ , где  $J_o(T_e) = \{j | \beta_j^o(T_e) > 0\}$  - подмножество единиц оборудования, которые на момент  $T_e$  могут быть использованы в ТСП;
- 3)  $R_o(T_e) \neq \emptyset$ , где  $R_o(T_e) = \{(d, k, z) | z = z_{d,k}^o(T_e); d \in D^o; k = 1, 2, \dots, n_d; t_{d,k,z}^o \leq T_e, (I_z \cap I_o(T_e)) = \emptyset, (J_z \cap J_o(T_e)) \neq \emptyset\}$  - подмножество операций из числа стоящих в очереди на обработку, для которых ранний срок запуска в обработку истек и имеются ресурсы, обеспечивающие их реализацию.

Условие 3 может быть модифицировано в условие 3а следующим образом. Обозначим через  $R_o'$  множество операций, сформированное по правилам условия 3. Тогда

$$R_o(T_e) = R_o' \cup R_o^e, \quad (2.1)$$

где  $R_o^e = \{(d, k, z) | z = z_{d,k}^e(T_e); d \in D^o; k = 1, 2, \dots, n_d; [I_z \cap I_o(T_e)] \neq \emptyset; [J_z \cap J_o(T_e)] \neq \emptyset; T_e < t_{d,k,z}^e \leq \min_{z \in R_o^e} (T_e + y_{d,k} \cdot t_z^e)\}$ .

По существу, модификация правил формирования множества  $R_o(T_e)$  позволяет расширить множество операций-претендентов, участвующих в производственной ситуации, с целью поиска наиболее эффективного решения.

Поскольку все другие состояния кроме производственной ситуации интереса не представляют и нет необходимости их рассматривать, то в дальнейшем состояние  $S(T_e)$  будет характеризовать наличие производственной ситуации на участке.

Рассмотрим процедуру принятия решений при наличии производственной ситуации  $S(T_e)$ . Реализация операций  $z \in R_o(T_e)$  в условиях ситуации  $S(T_e)$  возможна одним из допустимых ТСП:  $g \in G_z(T_e)$ ,  $G(T_e) = \{g | g = (d, k, z, i, j), i \in I_z \cap I_o(T_e); j \in J_z \cap J_o(T_e),$

$\alpha_i(T_e) \geq \beta_j$ . В результате для множества операций  $R_o(T_e)$  определено множество допустимых ТСП  $G(T_e, R_o) = \bigcup G_z(T_e)$ ,

элементы которого перенумеруем числами  $n=1, 2, \dots, N$ , т.е.  $G(T_e, R_o) = \{g_1, g_2, g_3, \dots, g_N\}$ . Присвоим порядковые номера  $m$  и элементам множеств  $I_o(T_e)$ ,  $J_o(T_e)$ ,  $R_o(T_e)$ ; где  $m=1, 2, \dots, m1$  - номера элементов  $i \in I_o(T_e)$ ,  $m=(m1+1), \dots, (m1+m2)$  -  $j \in J_o(T_e)$ ;  $m=(m1+m2+1), \dots, (m1+m2+m3)$  -  $z \in R_o(T_e)$ .

Пусть  $x_n = 0, 1$ ;  $n=1, 2, \dots, N$ , причем  $x_n = 1$  означает, что ТСП  $g_n \in G(T_e, R_o)$  решено включить в множество активных ТСП  $G^*(T_e)$ , а  $x_n = 0$  имеет обратный смысл. Переменная  $x_n$ ,  $n=1, \dots, (N+m3)$ , определяет число различных ТСП, которыми одновременно будет реализована операция  $z_m$  детали-партии  $(d, k)$ , где  $m=n-N$ . Область допустимых значений переменной  $x_n$ ,  $n > N$ , определяется целыми числами интервала  $[1, N_z]$ , где

$N_z$  - максимально допустимое число партий дробления детали-партии  $(d, k)$  на операции  $z$ . Тогда допустимое решение управляющего органа в ситуации  $S(T_e)$  включает ТСП  $g_n \in G(T_e, R_o)$ , соответствующие ненулевым компонентам вектора  $\bar{x}_n = \{x_1, x_2, \dots, x_N\}$ , который удовлетворяет следующим условиям:

$$\begin{aligned} \sum_{n=1}^N a_{m,n} x_n &\leq \alpha_{i_m}(T_e), m=1, 2, \dots, m1; \\ \sum_{n=1}^N a_{m,n} x_n &\leq \beta_{j_{m-m1}}^o(T_e), m=(m1+1), \dots, (m1+m2); \\ \sum_{n=1}^{N+m3} a_{m,n} x_n &= 0, m=(m1+m2+1), \dots, (m1+m2+m3). \end{aligned} \quad (2.2)$$

Элементы матрицы  $A = (a_{m,n})$  определены следующим образом:

- $a_{m,n} = \beta_j$ , если  $m=1, 2, \dots, m1$ ;  $g_n = (d, k, i_m, j) \in G(T_e, R_o)$ ;
- $a_{m,n} = 1$ , если  $m=(m1+1), \dots, (m1+m2)$  и  $g_n = (d, k, z, i, j_{m-m1}) \in G(T_e, R_o)$  или  $m=(m1+m2+1), \dots, (m1+m2+m3)$  и  $g_n = (d, k, z_{m-m1-m2}, i, j) \in G(T_e, R_o)$ ;
- $a_{m,n} = -1$ , если  $n > N$ ,  $n = N+m$ ,  $m=m1+m2+m$ ;
- $a_{m,n} = 0$  для всех  $m$  и  $n$ , не удовлетворяющих условиям а)-в).

Ясно, что в общем случае решение  $G^o(T_e)$  может быть получено не единственным образом, т.е. существует некоторое множе-



ство различных допустимых решений в производственной ситуации  $S(T_e)$ . Управляющий орган в этих условиях производит оценку приоритетности допустимых способов производства  $g_n \in G(T_e, R_e)$  и стремится к формированию наибольшего числа способов производства, выгодных с точки зрения полученных оценок приоритетности в данной производственной ситуации.

Если предположить, что из двух способов производства  $g_n$  и  $g_{\bar{n}}$  с оценками приоритетности  $b_n$  и  $b_{\bar{n}}$ ,  $b_n > b_{\bar{n}}$ , более выгодным является  $g_n$ , то критерий оптимальности допустимого решения  $G^*(T_e)$  в производственной ситуации  $S(T_e)$  можно записать следующим образом:

$$\max \sum_{n=1}^{N^*} b_n x_n, \quad (2.3)$$

где  $N^* = N + m$ ,  $b_n = b^*$  для  $n > N$ ,  $b^* \gg b_n$ ,  $n \leq N$ .

Достаточно большое число, которое используется для оценок способов производства  $g_n$ ,  $n > N$ , обеспечивает включение в решение  $G^*(T_e)$  наибольшего числа способов производства. Оценки  $b_n$ ,  $n \leq N$ , позволяют выбрать наилучшие из допустимых способов производства.

В имитационной модели оценки приоритетности способов производства  $g \in G(T_e, R_e)$  определяются по одному из правил из фиксированного набора  $\mathcal{X} = \{1, 2, \dots, \mathcal{X}^*\}$ . Каждому из правил  $\mathcal{X}_e \in \mathcal{X}$  соответствует определенный алгоритм, по которому на основе оценок элементов производства и их взаимосвязей в условиях производственной ситуации производится комплексная оценка приоритетности способов производства  $b_n$ . Введем оценки элементов производства и их взаимосвязей:  $b_n^1$  - оценки рабочих,  $b_n^2$  - единиц оборудования,  $b_n^3$  - операций,  $b_n^4$  - связей "операция - рабочий" и  $b_n^5$  - связей "операция-единица оборудования". Тогда расчет комплексной оценки  $b_n$ , определенной выше, осуществляется по формуле:  $b_n = M1 \cdot b_n^1 + M2 \cdot b_n^2 + M3 \cdot b_n^3 + M4 \cdot b_n^4 + M5 \cdot b_n^5$  (здесь  $M1, M2, \dots, M5$  - соответствующие коэффициенты нормирования). Набор правил  $\mathcal{X}$  в имитационной модели легко пополняется новыми. Этот факт может быть использован при конструировании правил принятия решений в календарном планировании производства конкретных участков.

Результируя вышеизложенное, можно сказать, что для разре-

шения производственной ситуации  $S(T_e)$  необходимо сформировать условия (2.2), произвести оценку приоритетности допустимых способов производства  $b_n, g_n \in G(T_e, R_o)$  по выбранному правилу  $\pi_e \in \pi$  и решить полученную задачу (2.2)-(2.3). Найденное решение задачи и определит  $G^o(T_e)$ .

В процессе расчета календарного план-графика для реальных производственных участков решение задач (2.2)-(2.3), которые возникают в производственных ситуациях  $S(T_e)$ , представляет большие технические трудности в связи с их значительной размерностью и целочисленным характером. Поэтому в предлагаемой имитационной модели решение (2.2)-(2.3) сведено к задаче нахождения максимального потока минимальной стоимости [5] на сформированной специальным образом сети. С практической точки зрения решение задач на сети более просто и эффективно реализуется на ЭВМ.

Разрешающая сеть  $\Gamma = (P, A)$  формируется на основе подмножеств элементов производства  $I_o(T_e), J_o(T_e), R_o(T_e)$  и подмножеств  $\{[I_z \cap I_o(T_e)], [J_z \cap J_o(T_e)], z \in R_o(T_e)\}$ , устанавливающих взаимосвязи между элементами производства. Множество дуг сети  $P$  состоит из подмножеств  $P_n, n = 1, 2, 3, 4, 5$ . Дуги  $p \in P_1$  взаимно-однозначно соответствуют элементам  $j \in S(T_e)$ ,  $p \in P_3$  - элементам  $z \in R_o(T_e)$ ,  $p \in P_5$  - элементам  $i \in I_o(T_e)$ . Каждая дуга  $p \in P_2$  отображает связь между операцией  $z \in R_o(T_e)$  и единицей оборудования  $j \in J_z \cap J_o(T_e)$ , на которой она может быть выполнена в условиях рассматриваемой производственной ситуации  $S(T_e)$ . Дуги этого подмножества смежны дугам  $p_1 \in P_1$  и  $p_3 \in P_3$ . Если обозначить  $p_i = (a_o, a_i)$ ,  $p_3 = (a_2, a_3)$ , где  $a_i \in A$  - вершины разрешающей сети, то дуги  $p_2 = (a_1, a_2) \in P_2$  имеют ориентацию от  $a_1$  к  $a_2$ . Аналогичным образом определяются дуги  $p_4 \in P_4$ , устанавливающие взаимосвязь между подмножествами дуг  $P_3$  и  $P_5$ . Дуги этого подмножества соответствуют взаимосвязям между операциями  $z \in R_o(T_e)$  и рабочими  $i \in I_z \cap I_o(T_e)$ , которые могут выполнить операцию  $z$  в условиях производственной ситуации  $S(T_e)$ . Ориентация дуг  $p_4 = (a_3, a_4) \in P_4$  направлена от  $a_3$  к  $a_4$ , где  $p_5 = (a_4, a^*) \in P_5$ . Завершая описание разрешающей сети  $\Gamma = (P, A)$ , отметим, что вершина  $a_o \in A$  - общая для всех дуг  $p_1 = (a_o, a_1) \in P_1$  и является источником сети. Вершина  $a^* \in A$  является общей для всех дуг  $p_5 = (a_4, a^*) \in P_5$  и определяет

сток разрешающей сети. Таким образом, каждый из путей, ведущих из источника сети  $a_0$  в сток  $a^*$ , однозначно определяет допустимый способ производства  $g_n \in G(T_e, R_0)$ . Если обозначить через  $b_p$  оценку дугового потока для  $p \in P$ , то, как и выше, комплексная оценка приоритетности способа производства  $g_n$  определяется через  $b_n = \sum_{p \in P(g_n)} b_p$ , где  $P(g_n)$  - дуги пути, соответствующего способу производства  $g_n$ . Здесь оценка  $b_p$  рассчитывается с учетом соответствующего коэффициента нормирования  $M_1, M_2, \dots, M_5$ .

Пропускные способности  $h_p$  дуг сети  $p \in P$  определяются следующим образом:

а)  $h_p = 1$  для  $p \in P_2 \cup P_4$ ;

б)  $h_p = \beta_j(T_e)$  для  $p \in P_1$ , где  $P$  соответствует  $j(p \rightarrow j)$ ,  $j \in J_0(T_e)$ ;

в)  $h_p = N_z$  для  $p \in P_3$ , где  $N_z$  - максимальное количество способов производства  $g_n$ , которые одновременно могут реализовывать детали-партию  $(d, k)$  на операции  $(d, k, z)$ , соответствующей дуге  $p$ ;

г)  $h_p = [\alpha_i(T_e) / \beta_j]$  для  $p \in P_5$  и  $p \rightarrow i$ , причем  $\beta_j$  определяется из того, что  $j \rightarrow p_1 \in P(g) = \{p_1, p_2, p_3, p_4, p_5\}$ .

Последнее условие предполагает, что рабочий  $i \in I$  обслуживает единицы оборудования  $j \in J$  с равными значениями параметра  $\beta_j$ . Практически для большинства производственных участков механообработки это условие выполняется.

Для того чтобы решить на сформированной сети  $\Gamma = (P, A)$  задачу нахождения максимального потока минимальной стоимости, введем дополнительную дугу  $p^* = (a^*, a_0)$  с достаточно большой пропускной способностью  $h_{p^*} = M$  и большой по абсолютной величине, но отрицательной по знаку оценкой дугового потока  $|b_{p^*}| \geq \max_{p \in P} b_p$ . Решение находим по алгоритму нахождения в сети циркуляций минимальной стоимости, предложенному Л.Фордом и Д.Фалкерсоном [5].

По найденному потоку в разрешающей сети  $\Gamma = (P, A)$  определяется множество путей из источника  $a_0$  в сток сети  $a^*$ , которые имеют ненулевые потоки. На основании этого множества формируется множество решений  $G^0(T_e)$ , принимаемых в условиях производственной ситуации  $S(T_e)$  управляющим органом.

Таким образом, в имитационной модели процесс разрешения производственной ситуации  $S(T_e)$  осуществляется через реализацию следующих процедур:

П1 - процедура формирования структуры разрешающей сети  $\Gamma = (P, A)$ ;

П2 - процедура расчета дуговых характеристик  $b_p, k_p, p \in P$ ;

П3 - процедура поиска оптимального потока в разрешающей сети согласно алгоритму [5] ;

П4 - процедура формирования множества решений  $g = (d, k, z, i, j) \in G^*(T_e)$  на основе найденного оптимального потока в разрешающей сети  $\Gamma = (P, A)$ .

Список основных процедур, реализованных в имитационной модели, включает также:

П5 - процедуру переформирования множества активных способов производства в соответствии с множеством решений  $G^*(T_e)$ ;

П6 - процедуру переформирования очереди на обработку  $R^*(T_e)$  по результатам реализации решений  $G^*(T_e)$ ;

П7 - процедуру анализа состояний элементов производства и определения очередного момента модельного времени  $T_{e+1}$  ;

П8 - процедуру завершения периода активизации способов производства  $g = (d, k, z, i, j) \in G^*(T_e)$ ;

П9 - процедуру формирования множеств элементов производства, определяющих производственную ситуацию  $S(T_e) = \{I_o(T_e), J_o(T_e), R_o(T_e)\}$ . Кроме перечисленных основных процедур имеется ряд дополнительных, которые обеспечивают различные модификации имитационной модели. Эти процедуры будут рассмотрены нами в следующем параграфе при описании модификаций модели.

Укрупненный алгоритм имитационной модели процесса календарного планирования дискретного производства, который лежит в основе различных модификаций модели, в операторном виде [6] может быть представлен следующим образом:

$$A_1, {}^{13/14} (P9)_2, P_{3+11}, (P1)_4, (P2)_5, (P3)_6, P_7^{11}, (P4)_8, \\ (P5)_9, (P6)_{10}, (P7)_{11}, P_{12}^{115}, P_{13+2}, (P8)_{14+2}, {}^{12} A_{15}, Я_{16}. \quad (2.4)$$

Здесь через  $(P1), \dots, (P9)$  обозначаются операторы алгоритма, реализующие соответствующие процедуры. Остальные операторы алгоритма выполняют следующие функции:

$A_4$  - ввод исходной информации по задаче календарного планирования и формирование начального состояния элементов производства и всей системы в целом  $S(T_0)$ ;

$P_5$  - проверка условия: "оформированные множества элементов производства  $I_0(T_e), J_0(T_e), R_0(T_e)$  порождают производственную ситуацию ?";

$P_7$  - проверка условия: "величина потока в разрешающей сети равна нулю ?";

$P_{12}$  - проверка условия: "период моделирования истек, т.е.  $T > T_N$  ?";

$P_{13}$  - проверка условия: "имеются в множестве  $G^*(T)$  активные способы производства  $g$ , для которых к моменту  $T$  период активизации истекает, т.е.  $t_g \leq T$  ?";

$A_{15}$  - обработка и выдача результата моделирования;

$A_{16}$  - конец моделирования.

После того, как нами рассмотрена принципиальная блок-схема имитационной модели, перейдем к описанию алгоритмов перечисленных выше процедур.

Описание алгоритмов процедур имитационной модели будем производить в виде операторных схем.

Алгоритм процедуры III:

$A_1, A_2, A_3, A_4, A_5, A_6, A_7^{20}, A_8, A_9^{16}, A_{10}, P_{11}^{15}$   
 $A_{12}, A_{13}, A_{14}, A_{15}^{11}, P_{16+19}, P_{17+19}, A_{18}, A_{19}^{17}, P_{20+7}, A_{21}, A_{22}$   
 $A_{23}, A_{24}^{35}, A_{25}, A_{26}^{31}, A_{27}, P_{28}^{130}, A_{29}, A_{30}^{23}, P_{31+26}, P_{32+34}, A_{33}$   
 $A_{34}, P_{35}^{124}, A_{36}, A_{37}, A_{38}, A_{39}, A_{40}, A_{41}, A_{42}.$

Отдельные операторы алгоритма выполняют следующие функции:

$A_1$  - формирование начальных значений:  $\pi_a^+ = 0, \pi_a^- = 0, a = a_0, a_0+1, \dots, a^*$ , здесь  $\pi_a^+$  - количество дуг, входящих в вершину с номером  $a$ ,  $\pi_a^-$  - количество дуг, выходящих из вершины  $a$ ;

$A_2$  - вычисление номеров вершин, характеризующих источник сети  $a_0 = 1$  и сток сети  $a^* = m_1 + 2 \cdot m_2 + m_3 + 2$ , где  $m_1, m_2, m_3$  - количество элементов в множествах  $J_0(T_e), R_0(T_e), I_0(T_e)$  соответственно;

$A_3$  - формирование дуги  $p_i = (a^*, a_0) \in P$  с номером  $n = I$ ;

$A_4$  - формирование дуг  $p_n = (a_0, n)$ ,  $n = 2, 3, \dots, (m+1)$ , где каждая дуга  $p_n$  соответствует единице оборудования  $j_{n-1} \in J_0(T_e) = \{j_1, j_2, \dots, j_m\}$ ;

$A_5$  - изменение значений параметров  $n_a^+ (a = a_0, a_0+1, \dots, a_0+m+1)$  и  $n_a^- (a = a^*, a_0)$  по результатам формирования дуг  $p_n$ ,  $n = 1, 2, \dots, (m+1)$ ;

$A_6$  - формирование начального значения счетчика просмотренных единиц оборудования  $j_m \in J(T_e) : m = 1$ ;

$A_7$  - идентификация выбранной единицы оборудования  $j = j_m$ ,  $j_m \in J_0(T_e)$ ;

$A_8$  - формирование начального значения счетчика  $k=1$  операций обработки  $z \in R_j$ , где  $R_j = \{z_1^j, z_2^j, \dots, z_{k(j)}^j\}$  - множество операций  $z \in R$ , которые могут быть выполнены на единице оборудования  $j$ ;

$A_9$  - идентификация операции, выбранной из множества  $R_j : z = z_k$ ;

$A_{10}$  - определения порядкового номера  $k1$  элемента  $z_{k1} \in R_0(T_e) = \{z_1^0, z_2^0, \dots, z_{m_0}^0\}$ , для которого  $z = z_{k1}$ ; равенство  $k1=0$  имеет место в случае, если  $z \notin R_0(T_e)$ ;

$P_{11}$  - проверка условия: " $k1 = 0$  ?";

$A_{12}$  - увеличение на единицу счетчика дуг  $p_n \in P : n = n+1$ ;

$A_{13}$  - формирование дуги  $p_n = (m+1, m+1+k1+1)$ , которая определяет возможность обработки операции  $z_{k1} \in R_0(T_e)$  на единице оборудования  $j_m \in J_0(T_e)$ ;

$A_{14}$  - изменение значений параметров, характеризующих количество входящих дуг в вершину с номером  $a = m+1+k1+1$  и выходящих дуг из вершины с номером  $\tilde{a} = m+1$ , т.е.  $n_a^+ = n_a^+ + 1$  и  $n_{\tilde{a}}^- = n_{\tilde{a}}^- + 1$ ;

$A_{15}$  - увеличение на единицу счетчика элементов множества  $R_j : k = k+1$ ;

$P_{16}$  - проверка условия: "все элементы множества  $R_j$  просмотрены?";

$P_{17}$  - проверка условия: " $n_{m+1}^- = 1$  ?";

$A_{18}$  - увеличение на единицу счетчика сформированных дуг сети:  $n = n+1$ ; формирование дуги  $p_n = (m+1, a^*)$ ; изменение значения количества выходящих из вершины  $a = m+1$  дуг:  $n_{m+1}^- = 1$  и числа входящих в  $a^*$  дуг:  $n_{a^*}^+ = n_{a^*}^+ + 1$ ;

$A_{19}$  - изменение значения счетчика просмотренных  $j \in J_0(T_e) : m = m+1$ ;

$P_{20}$  - проверка условия: "все элементы множества  $J_0(T_e)$  про-

смотрены?"

$A_{21}$  - анализ значений количества входящих дуг в вершины с номерами  $a = (m1+1), (m1+2), \dots, (m1+m2+1)$ ; для вершин  $a$ , у которых  $n_a^+ = 0$ , формируются дуги  $p_n = (a_0, a)$  с соответствующим изменением счетчика дуг  $n$ , значит, параметров  $n_a^+$  и  $n_a^-$ ;

$A_{22}$  - формирование дуг  $p_n = (m1+m+1; m1+m2+m+1)$ , соответствующих операциям  $z_m \in R_0(T_e)$ ,  $m = 1, 2, \dots, m2$ ; изменение значений счетчика  $n$ , параметров  $n_a^+$ , где  $a = m1+m2+m+1$ , и  $n_{\bar{a}}^-$ , где  $\bar{a} = m1+m+1$ ;

$A_{23}$  - формирование начального значения счетчика элементов множества  $R_0(T_e): m = 1$ ;

$A_{24}$  - идентификация операции обработки, выбранной из множества  $R_0(T_e): z = z_m$ ;

$A_{25}$  - формирование начального значения счетчика элементов из списка рабочих  $I_z$ , которые могут выполнить операцию  $z: k2 = 1$ ;

$A_{26}$  - идентификация выбранного элемента  $i_{k2} \in I_z$ ;

$A_{27}$  - определение номера  $k3$  элемента  $i_{k2}$  в множестве  $I_0(T_e)$ , для которого  $i_{k2} = i_{k3}$ ; если же  $i_{k2} \notin I_0(T_e)$ , то  $k3 = 0$ ;

$P_{28}$  - проверка условия: " $k3 = 0$ ?"

$A_{29}$  - формирование дуги  $p_n = (m1+m2+m+1; m1+2 \cdot m2+k3+1)$ , которая определяет возможность выполнения операции  $z_m \in R_0(T_e)$  рабочим  $i_{k2} \in I_0(T_e) \cap I_{z_m}$ ; изменение значения счетчика  $n$  и значений параметров  $n_a^+$  и  $n_{\bar{a}}^-$ , где  $a = m1+2 \cdot m2+k3+1$  и  $\bar{a} = m1+m2+m+1$ ;

$A_{30}$  - увеличение на единицу счетчика просмотренных элементов  $i_{k2} \in I_{z_m}: k2 = k2 + 1$ ;

$P_{31}$  - проверка условия: "все элементы множества  $I_{z_m}$  просмотрены?"

$P_{32}$  - проверка условия: " $n_a^- = 0, a = m1+m2+m+1$ ?"

$A_{33}$  - формирование дуги  $p_n = (m1+m2+m+1; a^*)$ ; изменение значения счетчика  $n$  и параметров  $n_a^-$ ,  $a = m1+m2+m+1$ , и  $n_{a^*}^+$ ;

$A_{34}$  - увеличение на единицу счетчика просмотренных элементов;

$P_{35}$  - проверка условия: " $m \leq m2$ ?"

$A_{36}$  - формирование дуг  $p_n = (m1+2 \cdot m2+m+1, a^*)$ , которые соответствуют элементам  $i_m$  множества  $I_0(T_e)$ ,  $m = 1, 2, \dots, m3$ , а также изменение при этом значения счетчика дуг  $n$ , числа входящих в  $a^*$  дуг  $n_{a^*}^+$  и числа выходящих из вершин  $a = m1+2 \cdot m2+$

$+m+1$  дуг  $n_a^-$ ;  
 $A_{37}$  - анализ значений параметров  $n_a^+$ ,  $a = (m+2 \cdot m+1) \dots (a^*+1)$ ;  
 формирование дуг  $p_n = (a_0, a)$  для тех вершин  $a$ , у которых  $n_a^+ = 0$ ; изменение значения счетчика  $n$  и параметра  $n_a^+$ ;  
 $A_{38}$  - переупорядочивание множества дуг  $P = \{p | p = (a_1, a_2)\}$  в соответствии с возрастанием номеров вершин  $a_1$ ;  
 $A_{39}$  - формирование списков  $P_a^-$  дуг, выходящих из вершины  $a$ ;  
 $A_{40}$  - переупорядочивание множества дуг  $P$  в соответствии с возрастанием номеров вершин  $a_2$ ,  $p = (a_1, a_2) \in P$ ;  
 $A_{41}$  - формирование списков  $P_a^+$  дуг, входящих в вершину  $a$ ;  
 $A_{42}$  - конец алгоритма процедуры П1.

Приведем некоторые пояснения описанного алгоритма. В процессе построения сети  $\Gamma = (P, A)$  каждое множество дуг  $P_m \in P$ ,  $m = 1, 2, \dots, 5$ , формируется отдельной группой операторов алгоритма. Параллельно с формированием дуг сети для каждой вершины  $a \in A$  ведется учет числа входящих и выходящих из нее дуг. Для вершин  $a$ ,  $p = (\bar{a}, a) \in P_1 \cup P_2$ , у которых  $n_a^- = 0$ , строятся фиктивные дуги  $p = (a, a^*)$ . Для вершин  $a$ ,  $p = (a, \bar{a}) \in P_3 \cup P_4$ ,  $n_a^+ = 0$ , вводятся фиктивные дуги  $p = (a_0, a)$ . Введение фиктивных дуг  $p = (a_0, a)$  и  $p = (a, a^*)$  обеспечивает связность сети  $\Gamma = (P, A)$ . Для того чтобы эти дуги не влияли на решение потоковой задачи, их пропускная способность приравнивается к нулю. В конце алгоритма структура сети  $\Gamma = (P, A)$  описывается через списки входящих и выходящих дуг для вершины сети реализацией операторов  $A_{38} - A_{41}$ .

Алгоритм процедуры П2:

$A_1 A_2 {}^9P_3 A_4 A_5 A_{6+8} A_7 A_8 {}^6P_{9+3} A_{10} A_{11} A_{12} {}^{20}P_{13} A_{14} A_{15} A_{16} A_{17+19}$   
 $A_{18} {}^{17}P_{19} A_{20+13} A_{21} P_{22} {}^{14}A_{23} A_{24} A_{25} A_{26} A_{27} A_{28} P_{29} {}^{424}A_{30} A_{31} A_{32} {}^{39}P_{33} A_{34} A_{35}$   
 $A_{36+38} {}^{34}A_{37} {}^{36}P_{39+33} A_{40} P_{41} {}^{431}A_{42} A_{43} A_{44} A_{45} A_{46} A_{47} A_{48} A_{49} A_{50} P_{51} {}^{143}A_{52} A_{53}$ .

Отдельные операторы алгоритма выполняют следующие функции:

$A_1$  - идентификация списка  $P_a^-$  дуг, выходящих из источника  $a_0$ ;  
 $A_2$  - формирование начального значения счетчика просмотренных дуг списка  $P_a^-$ :  $m=1$ ;  
 $P_3$  - проверка условия: " $a > m+1$ , где  $p_m = (a_0, a) \in P_a^-$  ?";  
 $A_4$  - идентификация единицы оборудования  $j_n \in J_0(T_e)$ , где  $n = a-1$ , соответствующей дуге  $p_m$ ;



- $A_5$  - формирование величины пропускной способности дуги:  
 $h_{p_m} = \beta_{jn}^o(T_e)$ ;  
 $A_6$  - вычисление оценки дугового потока  $\bar{b}_{p_m}$  в соответствии с выбранным правилом  $\pi_e \in \pi$  ;  
 $A_7$  - формирование параметров для фиктивной дуги  $p_m: h_{p_m} = 0, \bar{b}_{p_m} = 0$  ;  
 $A_8$  - изменение значения счетчика просмотренных дуг списка  $P_a^-: m = m + 1$  ;  
 $P_9$  - проверка условия: "все дуги списка  $P_a^-$  просмотрены?";  
 $A_{10}$  - формирование значения первого номера вершины  $a, p = (a, a1) \in P_e$  ;  
 $A_{11}$  - идентификация списка выходящих из вершин с номером  $a$  дуг  $P_a^- = \{p | p = (a, a1)\}$  ;  
 $A_{12}$  - формирование начального значения счетчика просмотренных дуг списка  $P_a^-: k1 = 1$  ;  
 $A_{13}$  - идентификация выбранной дуги  $p_{k1} = (a, \tilde{a}1) \in P_a^-$  ;  
 $P_{14}$  - проверка условия: " $\tilde{a}1 = a^*$  ?";  
 $A_{15}$  - идентификация операции  $z_{k2} \in R_o(T_e)$  , где  $k2 = \tilde{a}1 - m + 1$  ;  
 $A_{16}$  - оценка пропускной способности дуги  $p_{k1}: h_{p_{k1}} = 1$  ;  
 $A_{17}$  - вычисление оценки дугового потока  $\bar{b}_{p_{k1}}$  в соответствии с выбранным правилом  $\pi_e$  , которое характеризует эффективность назначения операции  $z_{k2}$  на единицу оборудования  $j_n \in J_o(T_e)$  ,  
 $n = a - 1$  ;  
 $A_{18}$  - формирование параметров для фиктивной дуги  $p_{k1} = (a, a^*)$  :  
 $h_{p_{k1}} = 0, \bar{b}_{p_{k1}} = 0$  ;  
 $A_{19}$  - изменение значения счетчика просмотренных дуг списка  $P_a^-$  :  
 $k1 = k1 + 1$  ;  
 $P_{20}$  - проверка условия: "все дуги списка  $P_a^-$  просмотрены?";  
 $A_{21}$  - увеличение номера вершины  $a$  на единицу:  $a = a + 1$  ;  
 $P_{22}$  - проверка условия: " $a \leq (m + 1)$  ?";  
 $A_{23}$  - формирование первого номера вершины  $a1$  такой, что  $P = (a1, a2) \in P_3: a1 = m + 2$  ;  
 $A_{24}$  - идентификация номера дуги  $k3: p_{k3} = (a1, \tilde{a}2) \in P_{a1}^-$  ;  
 $A_{25}$  - идентификация операции  $z_{k4} \in R_o(T_e)$  , где  $k4 = a1 - m + 1$  ;  
 $A_{26}$  - определение значения параметра  $N_{z_{k4}}$  , характеризующего возможное число партий, на которое допускается дробление дедтале-партии  $(d, k)$  на операции  $z_{k4}$  , и формирование про-

пропускной способности дуги  $P_{k3} : h_{P_{k3}} = N_{z_{k4}} ;$   
 $A_{27}$  - вычисление оценки дугового потока  $\bar{v}_{P_{k3}}$  в соответствии с правилом  $\pi_c$ , характеризующей приоритетность операции  $z_{k4}$ ;  
 $A_{28}$  - изменение значения номера просмотренной вершины:  $a1 = a1 + 1$ ;  
 $P_{29}$  - проверка условия: " $a1 \leq (m1 + m2 + 1)$  ?";  
 $A_{30}$  - формирование значения первого номера вершины  $a2$  такой, что  $p = (a2, a3) \in P_4 : a2 = m1 + m2 + 2$ ;  
 $A_{31}$  - идентификация списка выходящих из вершины с номером  $a2$  дуг  $P_{a2}^-$ ;  
 $A_{32}$  - формирование начального значения счетчика просмотренных дуг из списка  $P_{a2}^- : k1 = 1$ ;  
 $A_{33}$  - идентификация дуги  $P_{k1} = (a2, \bar{a}3) \in P_{a2}^-$ ;  
 $A_{34}$  - проверка условия: " $\bar{a}3 = a^*$  ?";  
 $A_{35}$  - формирование пропускной способности дуги  $P_{k1} : h_{P_{k1}} = 1$ ;  
 $A_{36}$  - вычисления оценки дугового потока  $\bar{v}_{P_{k1}}$  в соответствии с выбранным правилом  $\pi_c$ , характеризующим эффективность назначения на выполнение операции  $z_{k2} \in R_o(T_c)$ , где  $k2 = a2 - m1 - m2 - 1$ , рабочего  $i_{k3} \in I_o(T_c)$ , где  $k3 = \bar{a}3 - m1 - 2 \cdot m1 - 1$ ;  
 $A_{37}$  - формирование значений параметров для фиктивной дуги  $P_{k1} = (a2, a^*) : h_{P_{k1}} = 0, \bar{v}_{P_{k1}} = 0$ ;  
 $A_{38}$  - изменение значения счетчика просмотренных дуг из списка  $P_{a2}^- : k1 = k1 + 1$ ;  
 $P_{39}$  - проверка условия: "все дуги списка  $P_{a2}^-$  просмотрены?";  
 $A_{40}$  - изменение значения номера просмотренной вершины:  $a2 = a2 + 1$ ;  
 $P_{41}$  - проверка условия: " $a2 \leq (m1 + 2 \cdot m2 + 1)$  ?";  
 $A_{42}$  - формирование значения первого номера  $a3$  вершины, для которой  $p = (a3, a^*) \in P_5 : a3 = m1 + 2 \cdot m2 + 2$ ;  
 $A_{43}$  - идентификация дуги  $p = (a3, a^*) \in P_{a3}^-$ , выходящей из вершины с номером  $a3$ ;  
 $A_{44}$  - идентификация рабочего  $i = i_{k1}$ , где  $i_{k1} \in I_o(T_c)$ ,  $k1 = a3 - m1 - 2 \cdot m2 - 1$ ;  
 $A_{45}$  - идентификация списка дуг  $P_{a3}^+$ , входящих в вершину с номером  $a3$ ;  
 $A_{46}$  - идентификация списка дуг  $P_{A^*}^+$ , где  $A^* = \{\bar{a} / \bar{a} = \bar{a} - m2, (\bar{a}, a3) \in P_{a3}^+\}$ ;  
 $A_{47}$  - формирование значения параметра  $\beta_j$  нормативной степени занятости рабочего  $i$  на единицах оборудования  $j$ , где  $j = j_{k2} \in J(T_c)$  и  $k2 = \bar{a} - 1, (\bar{a}, \bar{a}) \in P_{A^*}^+$ ;

$A_{48}$  - определение величины пропускной способности  $h_p$  дуги  $p=(a_3, a^*)$ :  $h_p = [\alpha_i(T_e)/\beta_j]$ ;  
 $A_{49}$  - вычисление оценки дугового потока  $b_p$  в соответствии с выбранным правилом  $x_e$ , характеризующей приоритетность назначения на работу в данной производственной ситуации  $S(T_e)$  рабочего  $i = i_{e1} \in I_o(T_e)$ ;  
 $A_{50}$  - изменение значения номера просматриваемой вершины:  $a_3 = a_3 + 1$ ;  
 $P_{51}$  - проверка условия: " $a_3 < a^*$ ?";  
 $A_{52}$  - формирование параметров дуги  $p=(a^*, a_o)$ :  $h_p = M$ ,  $b_p = -\bar{M}$ , где  $M$  и  $\bar{M}$  - достаточно большие положительные числа;  
 $A_{53}$  - конец алгоритма процедуры П2.

В пояснение к алгоритму процедуры П2 можно сказать следующее. Как и в алгоритме процедуры П1, в данном алгоритме параметры  $h_p$  и  $b_p$  для каждого подмножества дуг  $P_m \subseteq P$ ,  $m = 1, 2, \dots, 5$ , формируются отдельными группами операторов независимо друг от друга. При реализации операторов  $A_6$ ,  $A_{17}$ ,  $A_{27}$ ,  $A_{36}$ ,  $A_{48}$  имеется в виду, что в модели правило  $x_e \in \mathcal{X} = \{x^1, x^2, \dots, x^k\}$  однозначно определяет алгоритм оценки каждого типа дуг  $P_m$ ,  $m = 1, 2, \dots, 5$ .

Алгоритм процедуры П3 взят из [5] и в данной работе не приводится.

Алгоритм процедуры П4:

$A_1, A_2, A_3, A_4, P_5^{119}, P_6^{119}, A_7, A_8, A_9, A_{10}, A_{11}$   
 $A_{12}, A_{13}, A_{14}, A_{15}, A_{16}, P_{17}^{121}, P_{18+18}, A_{19}, P_{20+24}, A_{21}^{17}$ .

Отдельные операторы алгоритма выполняют следующие функции:

$A_1$  - формирование начального значения счетчика способов производства в множестве  $G^o(T_e)$ :  $n = 0$ ;  
 $A_2$  - идентификация списка дуг  $P_{a_o}^-$ , выходящих из источника сети  $a_o$ ;  
 $A_3$  - формирование начального значения счетчика дуг из списка;  
 $A_4$  - идентификация  $m$ -й дуги из списка  $P_{a_o}^-$ :  $p_m = (a_o, a_1)$ ;  
 $P_5$  - проверка условия: " $a_1 > (m+1)$ ?";  
 $P_6$  - проверка условия: " $h_{p_m}^o = 0$ ?" ; здесь  $h_p^o$  - величина фактического дугового потока, найденного в результате решения оптимизационном потоковой задачи;  
 $A_7$  - идентификация списка дуг  $P_{a_1}^-$ , выходящих из вершины  $a_1$ ;

- $A_8$  - анализ величин фактических дуговых потоков  $h_{p_{k1}}^{\varphi}$ , где  $p_{k1} \in P_{a1}^-$ , с целью выделения дуги  $p_{k1} = (a1, a2)$ , для которой  $h_{p_{k1}}^{\varphi} > 0$ ;
- $A_9$  - идентификация дуги  $p_{k2} = (a2, a3) \in P_{a2}^-$ ;
- $A_{10}$  - идентификация списка дуг  $P_{a3}^-$ , выходящих из вершины  $a3$ ;
- $A_{11}$  - анализ величин фактических дуговых потоков  $h_{p_{k3}}^{\varphi}$ , где  $p_{k3} \in P_{a3}^-$ , с целью выделения дуги  $p_{k3} = (a3, a4)$ , для которой  $h_{p_{k3}}^{\varphi} > 0$ ;
- $A_{12}$  - идентификация дуги  $p_{k4} = (a4, a^*)$ ;
- $A_{13}$  - перерасчет величин дуговых потоков:  $h_p^{\varphi} = h_p^{\varphi} - 1$ , где  $p = p_m, p_{k1}, p_{k3}, p_{k4}$ ;  $h_{p^*}^{\varphi} = h_{p^*}^{\varphi} - 1$ ,  $p^* = (a^*, a_0)$ ;
- $A_{14}$  - увеличение на единицу счетчика сформированных способов производства  $g \in G^o(T_e)$ :  $n = n + 1$ ;
- $A_{15}$  - идентификация элементов производства:  $i_{n1} \in I_o(T_e), j_{n2} \in J(T_e), (a, k, z)_{n3} \in R_o(T_e)$ , где  $n1 = a4 - m1 - 2 \cdot m2 - 1, n2 = a1 - 1, n3 = a2 - m1 - 1$ ;
- $A_{16}$  - формирование способа производства  $g_n = [(a, k, z)_{n3}, i_{n1}, j_{n2}] \in G^o(T_e)$  и величины детали-партии  $y_{g_n} = [y_{a,k} / h_{p_{k2}}^{\varphi}]$ , реализуемой этим способом производства;
- $P_{17}$  - проверка условия: " $h_{p^*}^{\varphi} = 0$  ?";
- $P_{18}$  - проверка условия: " $h_{p_m}^{\varphi} = 0$  ?";
- $A_{19}$  - изменение значения счетчика просмотренных дуг списка  $P_{a^*}^-$ :  $m = m + 1$ ;
- $P_{20}$  - проверка условия: "все дуги списка  $P_{a^*}^-$  просмотрены?";
- $Я_{21}$  - конец алгоритма процедуры П4.

Описанный алгоритм процедуры формирования  $G^o(T_e)$  реализует поиск в сети  $\Gamma(P, A)$  путей из источника  $a_0$  в сток  $a^*$ , все дуги которых имеют ненулевые фактические дуговые потоки. По номерам вершин, инцидентным дугам найденных путей, идентифицируются элементы производства, которые обрезают способы производства, включаемые в множество решений  $G^o(T_e)$

Алгоритм процедуры П5:

$$A_1 \xrightarrow{17} A_2 \xrightarrow{18} A_3 \xrightarrow{19} A_4 \xrightarrow{20} P_{5+11} \xrightarrow{21} A_6 \xrightarrow{22} P_{7+12} \xrightarrow{23} A_9 \\ A_{10+13} \xrightarrow{24} A_{11} \xrightarrow{25} A_{12} \xrightarrow{26} A_{13} \xrightarrow{27} A_{14} \xrightarrow{28} A_{15} \xrightarrow{29} A_{16} \xrightarrow{30} P_{17+2} \xrightarrow{31} Я_{18}$$

Отдельные операторы алгоритма выполняют следующие функции:

$A_1$  - формирование начального значения счетчика реализованных решений  $g_n \in G^o(T_e)$ :  $n = 1$ ;

$A_2$  - идентификация способа производства  $g_n = (d, k, z, i, j) \in G(T_e)$  и элементов производства, определяющих этот способ производства;

$A_3$  - перерасчет параметров текущих состояний рабочего  $i$  и единицы оборудования  $j$ :  $\alpha_i(T_e) = \alpha_i(T_e) \cdot \beta_j$ ;  $\beta_j(T_e) = \beta_j(T_e) - t_j$ ;

$A_4$  - включение способа производства  $g_n \in G(T_e)$  в множество активных:  $G^*(T_e) = G^*(T_e) \cup g_n$ ;

$P_5$  - проверка условия: " $\delta_z = 0$  ?";

$A_6$  - расчет времени, требуемого по нормативам на обработку детали-партии  $y_{g_n}$  в поточном режиме:  $t = y_{g_n} \cdot t_z^H$ ;

$P_7$  - проверка условия: " $f_{d,k}(T_e) = 0$  ?";

$P_8$  - проверка условия: "предусматривается возможность сдвига срока начала периода активизации  $t_{g_n}^*$  способа производства  $g_n$  за счет перехода на параллельный вид движения детали-партии  $(d, k)$  на операции  $(d, k, z)$  ?";

$A_9$  - перерасчет срока готовности к началу обработки детали-партии на операции  $(d, k, z)$  с учетом перехода на параллельный вид движения:

$$t_{d,k,z}^z = t_{d,k,z}^z - \Delta t_1; \Delta t_1 = \min\{(y_{g_n} - 1) \cdot t_{z-i}^H; (y_{g_n} - 1) \cdot t_z^H\};$$

$A_{10}$  - определение срока начала периода активизации способа производства  $g_n$  при сдвиге срока готовности  $t_{d,k,z}^z$ :  $t_{g_n}^* = \max\{T_e, t_{d,k,z}^z, t_i, t_j\}$ , где  $t_i = \max\{t_g^* | g = (d, k, z, i, j) \in G(T_e, T_e)\}$ ;  $t_j = \max\{t_g^* | g = (d, k, z, i, j) \in G(T_e, T_e)\}$ ;

$A_{11}$  - определение времени, требуемого по нормативам на обработку детали-партии на операции  $(d, k, z)$  в партионном режиме:  $t = t_z^H$ ;

$A_{12}$  - определение срока начала периода активизации способа производства  $t_{g_n}^*$  в режиме, когда сдвиг периода активизации способа производства за счет смены вида движения не предусматривается:

$$t_{g_n}^* = \max\{t_{d,k,z}^z; T_e\};$$

$A_{13}$  - перерасчет дополнительных параметров, характеризующих текущее состояние рабочего  $i$  и единицы оборудования  $j$ :  $\tau_i^{np}(T_e) = \tau_i^{np}(T_e) + t_{g_n}^* - t_i^{ocb}$ ;  $c_i^b(T_e) = c_i^b(T_e) + y_{g_n} \cdot c_z^H$ ;  $\tau_j^{np}(T_e) = \tau_j^{np}(T_e) + t_{g_n}^* - t_j^{ocb}$ ;

$A_{14}$  - определение конца периода активизации способа производ-

ства  $g_n: t_{g_n}^6 = t_{g_n}^5 + t/2_i$ ;

$A_{15}$  - включение способа производства  $g_n$  и его характеристик  $t_{g_n}^5, t_{g_n}^6, \psi_{g_n}$  в план-график  $G: G(T_0, T_c) = G(T_0, T_c) \cup g_n$ ;

$A_{16}$  - увеличение значения счетчика реализованных решений  $g_n \in G^0(T_c): n = n + 1$ ;

$P_{17}$  - проверка условия: "все решения  $G^0(T_c)$  реализованы?";

$A_{18}$  - конец алгоритма процедуры П5.

В приведенном алгоритме процедуры П5 предусматривается возможность уплотнения план-графика производства за счет перехода на параллельный вид движения, если к моменту производственной ситуации  $S(T_c)$  и единица оборудования, и рабочий простаивают, ожидая работы. Остальные операторы алгоритма обеспечивают перерасчет параметров, характеризующих состояние элементов производства и способов производства  $G^0(T_c)$ , которые включаются в множество активных  $G^*(T_c)$ .

Алгоритм процедуры П6:

$A_1, A_2, A_3, P_4^{113}, A_5, P_6^{113}, A_{7+12}, P_8^{113}, P_9^{113}, A_{11+12}, A_{11}, A_{12}, A_{13}, P_{14+12}, A_{15}$ .

Отдельные операторы алгоритма выполняют следующие функции:

$A_1$  - формирование начального значения счетчика решений, реализованных в момент  $T_c: n = 1$ ;

$A_2$  - идентификация операции  $(d, k, z)$ , обрабатываемой в виде способа производства  $g_n = (d, k, z, i, j) \in G^0(T_c)$ ;

$A_3$  - исключение операции  $(d, k, z)$  из очереди на обработку  $R^*(T_c) = R^*(T_c) \setminus (d, k, z)$ ;

$P_4$  - проверка условия: "является операция  $(d, k, z)$  последней в технологии обработки заказа  $d$ , т.е.  $z = z_d^* \in R_d$ ?";

$A_5$  - формирование номера операции, следующей по технологии за назначенной на обработку  $(d, k, z): (d, k, z) = (d, k, z + 1)$ , и включение ее в очередь:  $R^*(T_c) = R^*(T_c) \cup (d, k, z)$ ;

$P_6$  - проверка условия: " $\delta_{z-1} = 0$ ?";

$A_7$  - формирование величины  $\Delta t$ , характеризующей период времени, раньше которого после назначения предыдущей операции на обработку из экономико-организационных условий нельзя начинать обработку на текущей операции  $(d, k, z)$ , для случая, если операции  $(z-1)$ ,  $z$  обрабатываются в партионном режиме:  $\Delta t = t_{z-1}^n$ ;

$P_8$  - проверка условия: " $\delta_z = 1$ ?";

$P_9$  - проверка условия: " $f_{d,k}(T_c) = 0$ ?";

$A_0$  - расчет величины  $\Delta t$  для случая, когда режим обработки на операции  $(z-1)$  поточный, а вид движения последовательный; расчет производится в соответствии с выражением (I.1), из которого следует  $\Delta t = (y_{g_n} - x_v(t_{g_n}^z)) \cdot t_{z-1}''$ , где  $v \in V_z^-$ ;

$A_H$  - расчет величины  $\Delta t$  при параллельном виде движения в соответствии с выражением (I.2), где  $\Delta t = \max\{0; \max t_{z-1}''; (y_{g_n} - x_v(t_{g_n}^z)) \cdot t_{z-1}'' - (y_{g_n} - 1) \cdot t_{z-1}'' - x_v(t_{g_n}^z) \cdot t_{z-1}''\}$ , где  $v \in V_z^-$ ;

$A_{12}$  - определение срока готовности к началу обработки операции  $(d, k, z) \in R^*(T_e): t_{d,k,z}^e = t_{g_n}^z + \Delta t$ ;

$A_{13}$  - изменение значения счетчика просмотренных решений  $g_n \in G^*(T_e): n = n + 1$ ;

$P_{14}$  - проверка условия: "все решения  $g_n \in G^*(T_e)$  просмотрены?";

$A_{16}$  - конец алгоритма процедуры П6.

алгоритм процедуры П7, в результате которой определяется очередной момент модельного времени  $T_e$ , включает последовательное выполнение трех операторов:

1) определение  $T' = \min\{t_{d,k,z}^e | (d, k, z) \in R^*(T_e)\}$ ;

2) определение  $T'' = \min\{t_g^e | g \in G^*(T_e)\}$ ;

3) определение момента модельного времени  $T_e = \min\{T', T''\}$ .

Из этого алгоритма следует, что к возникновению новой производственной ситуации может привести пополнение подмножества готовых к началу обработки операций или завершение периода активизации способов производства, в результате которого пополняются множества незанятых ресурсов.

Алгоритм процедуры П8, осуществляющий завершение периода активизации способов производства  $G^*(T_e)$ , имеет следующий вид:

$$A_1 \ A_2 \ P_{3+6} \ A_4 \ A_5 \ A_6 \ P_{7+8} \ A_8.$$

Отдельные операторы алгоритма выполняют следующие функции:

$A_1$  - формирование начального значения счетчика просмотренных активных способов производства:  $n = 1$ ;

$A_2$  - идентификация выбранного способа производства  $g_n = (d, k, z, i, j) \in G^*(T_e)$ ;

$P_3$  - проверка условия: " $t_{g_n}^e \leq T_e$  ?";

$A_4$  - перерасчет значений параметров текущего состояния рабо-

чего  $i$  и единицы оборудования  $j: \beta_j^*(T_e) = \beta_j^*(T_e) + 1; \alpha_i(T_e) = \alpha_i(T_e) + \beta_j;$

$A_5$  - исключение способа производства  $g_n$  из множества  $G^*(T_e): G^*(T_e) = G^*(T_e) \setminus g_n;$

$A_6$  - изменение значения счетчика просмотренных способов производства  $g_n \in G^*(T_e): n = n + 1;$

$P_7$  - проверка условия: "все способы производства  $G(T_e)$  просмотрены?";

$Y_8$  - конец алгоритма процедуры П8.

В пояснение этого алгоритма отметим: если множество  $G^*(T_e)$  упорядочено в соответствии с возрастанием значения параметра  $t_g^e, g \in G^*(T_e)$ , то при невыполнении условия  $P_3$  алгоритм заканчивает работу.

Алгоритм процедуры П9:

$A_1, A_2, A_3, P_{4+6}, A_5, Y_6.$

Отдельные операторы алгоритма выполняют следующие функции:

$A_1$  - формирование множества не занятых в момент  $T_e$  рабочих:  $I_o(T_e) = \{i | \alpha_i(T_e) > 0\};$

$A_2$  - формирование множества не занятых в момент  $T_e$  единиц оборудования:  $J_o(T_e) = \{j | \beta_j^*(T_e) > 0\};$

$A_3$  - формирование множества номеров операций из числа очередных, готовых к моменту  $T_e$  к назначению на обработку:  $R_o(T_e) = \{(d, k, z) | t_{d,k,z}^e \leq T_e; (d, k, z) \in R^*(T_e)\};$

$P_4$  - проверка условия: "предусматривается возможность расширения множества  $R_o(T_e)$  в данной производственной ситуации  $S(T_e)$  ?";

$A_5$  - формирование множества  $R'_o(T_e) \subseteq R^*(T_e) \setminus R_o(T_e)$  операций, дополнительно включаемых в соответствии с некоторым правилом (например, приведенном выше (2.1)) в множество претендентов на назначение на обработку:  $R_o(T_e) = R_o(T_e) \cup R'_o(T_e);$

$Y_6$  - конец алгоритма процедуры П9.

### § 3. Постановка экспериментов на имитационной модели

Как следует из описания имитационной модели в §§ 1,2, процесс календарного планирования производства в ней строится на



основе общей схемы путем использования правил из следующих трех множеств  $\mathcal{K}, \mathcal{F}, \mathcal{V}$ :

$\mathcal{K} = \{1, 2, \dots, K^*\}$  - правила оценки приоритетности способов производства в условиях производственных ситуаций  $S(T_e)$ ,  $e = 1, 2, \dots, N$ ;

$\mathcal{F} = \{1, 2, \dots, f\}$  - правила оценки вида движения деталей-партий в процессе обработки;

$\mathcal{V} = \{1, 2, \dots, v^*\}$  - правила отбора операций претендентов на назначение в обработку из числа очередных  $R^*(T_e)$  в условиях производственных ситуаций  $S(T_e)$ .

Вектор  $(\mathcal{K}_e, f, v)_T$ ,  $\mathcal{K}_e \in \mathcal{K}$ ,  $f \in \mathcal{F}$ ,  $v \in \mathcal{V}$ , однозначно определяет совокупность решений  $G^o(T)$ , реализуемых в производстве в момент  $T$ . Следовательно, стратегия деятельности управляющего органа в процессе календарного планирования определяется алгоритмом формирования для каждого момента  $T$  вектора  $(\mathcal{K}_e, f, v)_T$ . Вполне естественно возникает проблема: какие правила и с какой стратегией их выбора позволяют получить лучшее решение? Для того чтобы показать, каким образом на описанной в § 1,2 имитационной модели могут быть проведены данные исследования, остановимся на анализе алгоритмов выбора правил  $\mathcal{K}_e \in \mathcal{K}$ , т.е. правила  $f \in \mathcal{F}$  и  $v \in \mathcal{V}$  считаются зафиксированными.

Из вышесказанного следует, что для исследования различных стратегий выбора правил  $\mathcal{K}_e$  из предложенного набора  $\mathcal{K} = \{1, 2, \dots, K^*\}$  в ситуациях  $S(T_e)$  нужно выделить алгоритмы, моделирующие эти стратегии. Выделенные алгоритмы оформляются в виде отдельных процедур. Так как моделирующий алгоритм построен по модульному принципу, то процедура выбора правил без труда может быть в него включена.

Если в процедуре выбора правила  $\mathcal{K}_e \in \mathcal{K}$  в ситуации  $S(T_e)$  не используется информация о текущих характеристиках состояния элементов производства и совокупного состояния в целом, то процедура ставится в начале моделирующего алгоритма, и в результате ее реализации получается последовательность  $\{\mathcal{K}_e\}_{e=1}^N$ . В процессе моделирования номера используемых правил выбираются из данной последовательности  $\{\mathcal{K}_e\}_{e=1}^N$ . В случае, когда процедура выбора правила  $\mathcal{K}_e \in \mathcal{K}$  требует оценки ситуации  $S(T_e)$ , то оператор  $(P(\mathcal{K}_e))$ , реализующий эту процедуру, вставляется

в алгоритм (2.4) перед оператором  $(\Pi 2)_5$ . Номер правила  $\pi_{\ell} \in \pi$ , используемого при разрешении ситуации  $S(T_{\ell})$ , в этом случае определяется на каждом этапе моделирования.

Перейдем к описанию некоторых алгоритмов выбора правил, которые реализованы в имитационной модели.

### 3.1. Фиксированный выбор правил $\pi_{\ell} \in \pi$ .

Данный подход не требует дополнительных процедур, и алгоритм моделирования, описанный в § 2, практически не изменяется. Здесь последовательность правил  $\{\pi_{\ell}\}_{\ell=1}^N$  задается исследователем и в процессе моделирования не изменяется. В частности, если  $\pi_1 = \pi_2 = \dots = \pi_N$ , то на имитационной модели можно провести исследования влияния каждого из правил на возможные параметры оценки деятельности участка и их траектории в процессе производства. При необходимости получить лучший, в смысле выбранного критерия  $F$ , план-график производства по данной модификации модели последовательно проигрывается каждое правило  $\pi_{\ell} \in \pi$  и осуществляется выбор на основании сравнения значений  $F(G_{\pi_{\ell}})$ .

### 3.2. Алгоритм случайного ненаправленного формирования последовательности $\{\pi_{\ell}\}$ .

При данной стратегии формирование последовательности правил  $\{\pi_{\ell}\}_{\ell=1}^N$  осуществляется на основе случайного выбора правила  $\pi_{\ell}$  с вероятностью  $p_{\pi}$ , где  $\sum_{\pi \in \pi} p_{\pi} = 1$ . Численная

реализация процедуры выбора правила  $\pi_{\ell}$  осуществляется следующим образом. Интервал  $[0, 1]$  разбиваем на интервалы  $\Delta_1 = [0, p_1]$ ,  $\Delta_2 = (p_1, p_1 + p_2), \dots, \Delta_{\sum_{\pi \in \pi} p_{\pi}} = (\sum_{\pi \in \pi} p_{\pi}, 1]$ . Для каждого  $\ell$ -го шага гене-

рируется случайное число  $\xi$  с равномерным распределением на интервале  $[0, 1]$  и осуществляется выделение интервала  $\Delta_{\pi}$ , для которого  $\xi \in \Delta_{\pi}$ . В результате этой процедуры правило с номером  $\pi$  принимается в качестве  $\ell$ -го элемента последовательности, т.е.  $\pi_{\ell} = \pi$ . Для предложенного алгоритма осуществляется  $n$ -кратное проигрывание имитационной модели, в результате которого получают план-графики производства  $G_k$ ,  $k = 1, 2, \dots, n$ , со значениями критерия оценки оптимальности  $F(G_k)$ . Из этого множества выбирается план-график  $G$ , для которого  $F(G) = \min_k F(G_k)$  (или  $\max_k F(G_k)$ ).

Ясно, что при таком подходе информация о результатах предыдущих проигрываний имитационной модели никаким образом не влияет на формирование последовательности в текущем эксперименте.

3.3. Алгоритм случайного выбора правила  $\pi_l \in \mathcal{X}$  с обучением.

На практике при решении сложных задач человеком большое значение имеет накопление опыта, которое схематично можно представить в виде последовательности этапов: "эксперимент-анализ результатов эксперимента - корректировка своих представлений о последствиях применения использованных в эксперименте правил - эксперимент". Эту идею можно заложить в имитационную модель в виде процесса обучения, общая схема которого представляется следующим образом.

Пусть  $P_{\pi_l, l}^k$  - вероятность выбора правила  $\pi_l$  в условиях производственной ситуации  $S(T_l)$  для  $k$ -го проигрывания имитационной модели, где  $\sum_{l=1}^L P_{\pi_l, l}^k = 1$ ,  $l = 1, 2, \dots, L = 0, 1, 2, \dots$ . Набор  $P_{\pi_l, l}^0$ ,  $\pi_l \in \mathcal{X}$ ,  $l = 1, 2, \dots$ , задает начальные значения вероятностей выбора правил перед началом процесса обучения. Оценка результата  $k$ -го проигрывания производится на основе сравнения значения выделенного параметра  $F(G)$ , характеризующего качество план-графика  $G^k$ , со значением  $F(G^*)$  эталонного план-графика  $G^*$ . По результатам оценки, в соответствии с принятым алгоритмом обучения, производится перерасчет вероятностей выбора правил  $\{P_{\pi_l, l}^{k-1}\} \rightarrow \{P_{\pi_l, l}^k\}$ , на основе которых генерируется последовательность  $\{\pi_l^k\}_{l=1}^N$  для  $(k+1)$ -го проигрывания.

Как показывает опыт подобных исследований [7], моделирующие процесс обучения алгоритмы различаются:

- а) по методу выбора эталонного план-графика  $G^*$  для оценки результата  $k$ -го проигрывания;
- б) по методу оценки качества полученного в  $k$ -м проигрывании план-графика  $G^k$  и алгоритму переоценки вероятностей выбора правил  $\{P_{\pi_l, l}^{k-1}\}$ .

Для исследования можно предложить несколько различных моделирующих процесс обучения алгоритмов, которые оформляются в виде отдельных процедур и включаются в имитационную модель,

описанную в § 2. Для примера рассмотрим алгоритмы, которые исследовались на имитационной модели.

Алгоритм 1. На начальном этапе за эталонный  $G^3$  принимается план-график, который является лучшим в смысле выделенного параметра  $F(G)$  среди полученных методом фиксированного выбора правил для исследуемого набора правил  $\mathcal{X}$ . После  $k$ -го проигрывания фиксируется дополнительно  $G_k^3$ , для которого

$$F(G_k^3) = \min \{F(G_{k-1}^3), F(G_k^3)\}; G_k^3 = G^3, \text{ если } k = 1.$$

Начальные значения  $P_{x_l}^0$  определяют равновероятный выбор каждого правила  $x_l$  из исследуемого набора правил  $\mathcal{X}$ . После  $k$ -го проигрывания перерасчет вероятностей выбора правил производится в соответствии с выражениями:

$$\begin{aligned} P_{x_l, l}^k &= (P_{x_l, l}^{k-1} + \alpha) / \sum_{i=1}^n P_{i, l}^{k-1} + \alpha && \text{для } x_l = x_l^{k-1}, \\ P_{x_l, l}^k &= P_{x_l, l}^{k-1} / (\alpha + \sum_{i=1}^n P_{i, l}^{k-1}) && \text{для } x_l \neq x_l^{k-1}. \end{aligned} \quad (3.1)$$

Значение параметра  $\alpha$  определяется следующим образом:

$$\alpha = \begin{cases} \alpha_0 \cdot P_{x_l, l}^{k-1}, & \text{если } F(G^k) \geq F(G_k^3) \geq F(G_{k-1}^3); \\ \alpha_0(\alpha_0 + 1) \cdot P_{x_l, l}^{k-1}, & \text{если } F(G_{k-1}^3) > F(G^k); \\ -\alpha_0 \cdot P_{x_l, l}^{k-1}, & \text{если } F(G^k) > F(G^3). \end{cases} \quad (3.2)$$

Величины  $\alpha_0$  и  $\alpha_1$  задаются перед началом эксперимента и в процессе обучения на меняются. Величина  $\alpha_0$  характеризует нормативную степень изменения вероятностей, а  $\alpha_1$  характеризует степень поощрения за получение наилучшего результата. Ясно, что, изменив в (3.2) определение значения  $\alpha$  следующим образом:  $\alpha = 0$ , если  $F(G^k) > F(G^3)$ , получим модификацию описанного алгоритма обучения. Содержательно это отражает такой подход к обучению, когда учитывается только положительный опыт. Численная реализация алгоритма генерирования последовательностей правил  $\{x_l^k\}_{l=1}^n$  на основе значений вероятностей  $P_{x_l, l}^k$  осуществляется аналогично тому, как это было описано для алгоритма случайного выбора правил (п.3.2).

Алгоритм 2. Моделирование процесса обучения в этом algo-

ритме несколько отличается от предыдущего тем, что рассматриваются вероятности  $P_{x,i,l}^k$  перехода от правила с номером  $x$ , которое было использовано при разрешении ситуации  $S(T_{l-1})$ , к правилу с номером  $i$ , которое будет использовано в ситуации  $S(T_l)$ . Таким образом, перед  $k$ -м проигрыванием имеем последовательность матриц  $\{P_{l-1}^{k-1}\}_{l=1}^N$ , где  $P_{l-1}^{k-1} = (P_{x,i,l-1}^{k-1}, \dots, P_{x,\pi^*,l-1}^{k-1})$  - матрица из одного столбца, элементы которого характеризуют вероятность выбора правила  $x$  в ситуации  $S(T_{l-1})$ ,  $l=1$ , а  $P_l^{k-1} = (P_{x,i,l}^{k-1})$ ,  $l > 1$ , - матрица  $\pi^* \times \pi^*$ , для которой  $\sum_{x=1}^{\pi^*} P_{x,i,l}^{k-1} = 1$ ,  $i=1, 2, \dots, \pi^*$ . Начальное состояние значений элементов матриц  $\{P_l^0\}_{l=1}^N$  определяет равновероятный выбор правил в последовательности  $\{x_l^1\}_{l=1}^N$ .

В качестве эталонного план-графика  $G^0$  для начала эксперимента выбирается лучший среди полученных методом фиксированного выбора правил для исследуемого набора  $\pi$ . В процессе обучения после  $k$ -го проигрывания за эталонный план-график выбирается лучший из полученных за все предыдущие проигрывания, т.е. для которого  $F(G^0) = \min_{i \leq k} \{F(G^i), F(G^0)\}$ .

Перерасчет значений элементов матриц по результатам  $k$ -го проигрывания осуществляется согласно

$$\begin{aligned} P_{x,i,l}^k &= (P_{x,i,l}^{k-1} + \alpha) / (\alpha + \sum_{j=1}^{\pi^*} P_{x,j,l}^{k-1}), \text{ если } x = x_{l-1}^{k-1}, i = x_{l-1}^{k-1}; \\ P_{x,i,l}^k &= P_{x,i,l}^{k-1} / (\alpha + \sum_{j=1}^{\pi^*} P_{x,j,l}^{k-1}), \text{ если } x = x_{l-1}^{k-1}, i \neq x_{l-1}^{k-1}; \\ P_{x,i,l}^k &= P_{x,i,l}^{k-1}, \text{ если } x \neq x_{l-1}^{k-1}. \end{aligned} \quad (3.3)$$

В отличие от предыдущего алгоритма обучения в данном расчет величины  $\alpha$  производится следующим образом:

$$\alpha = \alpha_0 \cdot (F(G^0) - F(G^k)) / F(G^0); \quad (3.4)$$

величина  $\alpha_0$  задается перед началом процесса обучения и не меняется в течение всего эксперимента. По аналогии с предыдущим, если учитывать только положительный опыт, то расчет  $\alpha$  производится так:

$$\alpha = \begin{cases} 0 & , \text{ если } F(G^k) > F(G^0); \\ \alpha_0 \cdot (F(G^0) - F(G^k)) / F(G^0) & , \text{ если } F(G^k) \leq F(G^0). \end{cases}$$

При исследовании процессов обучения на основе приведенных нами двух алгоритмов нетрудно получить их модификации, учитывающие особенности конкретного типа обучения.

#### 3.4. Выбор правил $\mathcal{X}_e \in \mathcal{X}$ в диалоговом режиме.

Описанные выше методы формирования последовательности  $\{\mathcal{X}_e\}_{e=1}^N$  правил, используемых при разрешении производственных ситуаций  $\mathcal{S}(T_e)$ , опирались на идею случайного поиска (не-направленного и направленного). Естественно, при таком подходе практически не учитывается предистория использования правил в рамках одной реализации эксперимента в предыдущих производственных ситуациях  $\mathcal{S}(T_i)$ ,  $T_i < T_e$ , а также характеристика сложившейся в момент  $T_e$  ситуации  $\mathcal{S}(T_e)$ , которая требует разрешения. Учитывая сложность решаемых задач, данный недостаток может быть устранен, если в имитационную модель включить человека, которого далее будем именовать ЛПР (лицо, принимающее решение). Для реализации такого подхода набор основных процедур, на базе которых формируется моделирующий алгоритм (§ 2), пополняется процедурой, обеспечивающей диалог между имитационной моделью и ЛПР. Данная процедура должна обеспечить вывод через диалоговое устройство информации, характеризующей состояние элементов производства в выделенный момент времени  $T$  и состояние объекта управления в целом, и ввод в имитационную модель решений, принимаемых ЛПР. По существу, включение диалоговой процедуры в имитационную модель позволяет рассматривать ее как базу для организации деловой игры. В рамках организованной на базе данной модификации имитационной модели деловой игры открываются возможности для более глубокого анализа процессов календарного планирования и, в частности, исследования комбинирования различных правил  $\mathcal{X}_e \in \mathcal{X}$  в процессе построения план-графика производства.

Таким образом, из вышеизложенного следует: чтобы провести исследование выбранного аспекта процесса календарного планирования с нужной глубиной, достаточно модифицировать основ-

ной алгоритм имитационной модели, предложенный в § 2, путем разработки и включения в модель дополнительных процедур. Вполне естественно, что предложенные в данной работе модификации модели не исчерпывают ее возможностей и могут быть расширены.

Результаты экспериментальных исследований будут изложены в отдельной работе.

#### ЛИТЕРАТУРА

1. МИРОНОСЕЦКИЙ Н.Б. Экономико-математические методы календарного планирования. Новосибирск, "Наука", 1973.
2. ПОРТУГАЛ В.М., СЕМЕНОВ А.И. Теория расписаний. М., "Знание", 1972.
3. ПЕТРОВ В.А. Групповое производство и автоматизированное оперативное управление. Л., "Машиностроение", 1975.
4. ТАНАЕВ В.С., ШКУРБА В.В. Введение в теорию расписаний. М., "Наука", 1975.
5. ФОРД Л.Р., ФАЛКЕРСОН Д.Р. Потоки в сетях. М., "Мир", 1966.
6. ЛЯПУНОВ А.А. О некоторых общих вопросах кибернетики. - В кн.: Проблемы кибернетики, вып. I, 1958, с.5-22.
7. ФИШЕР Г., ТОМПСОН Г.Л. Комбинация локальных правил календарного планирования применительно к самонастраивающимся на вероятностной основе программам для электронно-вычислительной машины. - В кн.: Календарное планирование. М., "Прогресс", 1966, с. 260-291.

Поступила в ред.-изд. отд.  
II.XI.1977 г.