УДК 681.5:621

Д.Э. Лысенко

Одеський національний політехнічний університет, м. Харків

МОДЕЛИ СБОРОЧНЫХ РАБОТ В ЕДИНИЧНОМ И МЕЛКОСЕРИЙНОМ ПРОИЗВОДСТВЕ

Работа посвящена планированию производственных процессов в сборочном производстве. Рассмотрена задача формирования расписаний работ рабочих мест, обеспечивающего выполнение производственной программы с наименьшими затратами машинного времени при наличии ограничений на общую загрузку машин. Построена математическая модель задачи. Выполнен анализ свойств предложенной математической модели. Предложен алгоритм её решения. Теоретические результаты проиллюстрированы примером формирования расписания.

Ключевые слова: *сборочное производство, теория расписаний, математическое моделирование*

Введение

Существует несколько распространённых схем организации работ в сборочном производстве сложной наукоёмкой продукции (стапельная сборка). Одна из схем организации работ предусматривает неподвижный объект сборки и перемещение исполнителей работ в соответствии с требованиями технической документации. В другом варианте объект сборки перемещается последовательно к стационарным сборочным площадкам (стапелям), где выполняется плановый объём работ, определённых для этого специализированного рабочего места (участка). Различные схемы организации работ требуют различных подходов к планированию и разных критериев оптимизации управленческих решений, одним из которых является снижение загрузки рабочих мест с целью более эффективного использования производственного оборудования [1].

Планирование и оценка реализуемости сборочных технологий возможна на основе разработки соответствующих математических моделей организации стапельных работ. В общей постановке задача может рассматриваться как определение наилучшего графика выполнения m независимых операций на n различных по оснащенности рабочих местах («машинах»), если заданы времена выполнения i-й операции на k-й машине (участке). Время работы по выполнению операций, а также матрица времени перестройки (переналадки) каждого участка с выполнения i-й работы на выполнение j-й задачи определены технологическим маршрутом [2].

Постановка задачи

В формальной постановке задача может быть сформулирована следующим образом. На n различных по оснащённости (специализации) рабочих местах необходимо выполнить m независимых работ. Будем считать заданными матрицу времён выполнения i-й операции на k-м рабочем месте $A=\|a_{ik}\|$ ($i=1,2,...,m$, $k=1,2,...,n$); матрица времён переналадки k-го рабочего места с выполнения i-й на j-ю операцию $B^k=\|b_{ij}^k\|$ ($i,j=1,2,...,m$, $k=1,2,...,n$). Также будем считать заданной матрицу $D=\|d_{ik}\|$ ($i=1,2,...,m$, $k=1,2,...,n$). В соответствии с условиями эксплуатации k-я машина должна останавливаться для техобслуживания через временные интервалы T_{ks} ($S=1,2,...,P$). На основе сформированных постановок можно строить модели планирования и контроля реализации планов развития предприятия и определить расписание работ участков, обеспечивающее выполнение плановых работ с минимальным расходом машинного времени при условии, что общее время занятости k-го рабочего места не будет превышать G_k.

Далее будем считать, что все величины a_{ik}, b_{ij}^k, d_{ik}, $i,j=1,2,...,m$, $k=1,2,...,n$; T_{ks}, θ_{ks}, G_k; $S=1,2,...,P$ являются целыми числами. Приведенные задачи могут быть сведены к задачам целочисленного программирования путём введения переменных со значениями «0» и «1».

Поставим в соответствие каждой операции $i,j=1,2,...,m$, которая выполняется на рабочем месте ($k=1,2,...,n$), вершину i графа JU и обозначим её i^k. Изображаем дугу $\left(i^k, j^k\right)$ со стрелками, направленными от i^k к j^k, длина которой $\left(c_{i^k j^k}\right)$ равна b_{ij}^k. Вершины $(m+1)^k$, отражающие переналадку рабочих мест, соединяем с вершинами i^k ($i=1,2,...,m$, $k=1,2,...,n$) дугами, направленными от $(m+1)^k$ к i^k, длины их $\left(c_{(m+1)^k j^k}\right)$ равны d_{ik}, и дугами, противоположно направленными, длина которых равна нулю.

Вершины i^k ($i=1,2,...,m$, $k=1,2,...,n$) охватываем контуром, длина которого $\left(c_{i^k j^k}\right)$ равна a_{ik}, а вершины $(m+1)^k$, $k=1,2,...,n$ – контурами единичной длины. Каждому плану работы k-й машины ставится в соответствие некоторый путь в k-м графе. Обозначим длину этого пути l_k, а множество вершин, по контурам которых проходит искомый путь, – f^k. Тогда определение расписания, обеспечивающего выполнение производственной программы, сводится к определению таких путей в графах, начинающихся в вершинах $(m+1)^k$, $k=1,2,...,n$, которые удовлетворяют следующим требованиям:

1) когда длина пути в k-м графе становится равной $h^k \le T_{ks}$ ($k=1,2,...,n$, $s=1,2,...,n$), искомый путь через дуги нулевой длины должен возвратиться в вершину $(m+1)^k$ k-го графа и, достигнув длины $h^k = T_{ks}$, θ_{ks} раз пройти по петле, охватывающей эту вершину;

2) для всех допустимых путей, обеспечивающих выполнение производственной программы, справедливо соотношение

$$\bigcup_{k=1}^{n} f^k = J\,;\ \varphi^k \le 1\,;\ k=1,2,...,n\,, \qquad (1)$$

где $J:\{1,2,...,m\}$ – множество всех операций, выполняемых на машинах; φ^k – множество путей в k-м графе, которые входят в допустимое расписание;

3) длина пути в k-м графе должна быть меньше или равна G_k (если не накладывается ограничения на общее время k-й машины, то $G_k = \infty$).

Задача сводится к определению среди допустимых путей в графах таких, чтобы

$$\sum_k l_k \to \min\,, \qquad (2)$$

Исследование свойств задачи

Алгоритмы решения задач используют специфику графов. Как правило, на практике дуги графов удовлетворяют соотношениям

$$c_{ij} \le c_{ip} + c_{pj}\,;\ i, p, j \in J \qquad (3)$$

Пусть

$$J_1 \subset J_2\,;\ (m+1) \in J_2 \cap J_1\,.$$

Обозначим соответственно длину кратчайшего пути, начинающегося в вершине $i=(m+1)$ и проходящего по контурам каждой из вершин $i \in J_1$ графа, – через l_1, а кратчайшего пути, заходящего, кроме того, ещё в каждую из вершин графа $J_3 = J_2 \setminus J_1$, – через l_2.

Лемма 1. Если дуги графа удовлетворяют соотношениям (4), то

$$l_1 \le l_2\,. \qquad (4)$$

Доказательство. Предположим, что

$$J_3 = J_2 \setminus J_1 = j_m\,.$$

Пусть последовательность вершин, соответствующая пути длиной l_1, имеет вид

$$\left\{j_{m+1}, j_{p_1}, j_{p_1}, j_{p_2}, j_{p_2}, ..., j_{p_{m-1}}, j_{p_{m-1}}\right\},$$

а пути диной l_2 –

$$\left\{j_{m+1}, j_{t_1}, j_{t_1}, ..., j_m, ..., j_{t_{m-1}}, j_{t_{m-1}}\right\}$$

или

$$\left\{j_{m+1}, j_{t_1}, j_{t_1}, j_{t_2}, ..., j_{t_{m-1}}, j_{t_{m-1}}, j_m\right\}.$$

Построим путь, длину которого обозначим l_1^*, удовлетворяющий условиям леммы и проходящий через вершины множества J_1 в последовательности

$$\left\{j_{m+1}, j_{t_1}, j_{t_1}, j_{t_2}, ..., j_{t_{m-1}}, j_{t_{m-1}}\right\}.$$

Очевидно, $l_1 \le l_1^*$, так как l_1 – длина кратчайшего пути. Сравнивая последовательности, соответствующие путям длиной l_2 и l_1^*, приходим к выводу, что $l_1^* \le l_2$ в силу (4) либо того, что $c_{j_{t_{m-1}}} \ge 0$. Следовательно,

$$l_1 \le l_2\,,$$

что доказывает лемму.

Следствия леммы:

1. При построении допустимых расписаний можно ограничиться рассмотрением вариантов, в которых каждый путь, зашедший в вершину t^k k-го графа, не может быть продолжен в вершину j^k до тех пор, пока не пройдёт по петле при вершине i^k.

2. Оптимальное расписание может быть найдено среди путей, подмножества вершин, по

петлям, при которых проходят искомые пути и которые удовлетворяют соотношениям

$$f^{k_1} \bigcap f^{k_2} = \varnothing; \; k_1, k_2 = 1, 2, ..., n. \qquad (4')$$

Принимая во внимание первое следствие леммы, исходный граф может быть заменён эквивалентным графом без петель при вершинах, дуги которого удовлетворяют соотношениям

$$\alpha_{i^k j^k} = c_{i^k j^k} + c_{j^k j^k}; \; i, j = 1, 2, ..., m+1; \quad k = 1, 2, ..., n. \qquad (5)$$

Алгоритм решения задачи

Вычислительные схемы точного метода решения задачи строятся на основе метода последовательного анализа вариантов [3] и состоят из следующих этапов: построение в k-м графе всех путей длиной G_k, которые могут входить в допустимое расписание, с отсевом «неперспективных», построение всех допустимых расписаний работы $2, 3, ...n$ машин, построение расписаний, удовлетворяющих (1), и выбор среди них оптимального.

Алгоритм определения допустимых путей длиной G_k в каждом графе заключается в последовательном построении в k-м графе путей длиной

$$L_{1k}, L_{1k} + L_{2k}, ..., L_{1k} + L_{2k} + ... + L_{sk},$$

начинающихся в вершинах $(m+1)^k$, $k = 1, 2, ..., n$, где

$$L_{ks} = T_{ks} - \left(T_{k(s-1)} + \theta_{k(s-1)}\right); \quad s = 1, 2, ..., r; \; k = 1, 2, ..., n. \qquad (6)$$

По мере построения производится отсев неперспективных путей.

Построение производится процедурой пометок вершин графа, которая позволяет получить компактные вычислительные схемы, удобные для программой реализации [4, 5].

Метка вершины t^k состоит из следующих признаков:

$$r_t^k = \left(r_t^{1k}, r_t^{2k}, r_t^{3k}\right); \; f_t^k; \; l_t^k, \qquad (7)$$

где r_t^{1k} – порядковый номер пути, заходящего в вершину t^k; r_t^{2k} – номер предыдущей вершины этого пути; r_t^{3k} – порядковый номер этого пути для предыдущей вершины; f_t^k – множество вершин, по петлям которых проходит искомый путь; l_t^k

Вначале строим все допустимые пути в первом графе ($k=1$), затем во втором, третьем и т.д. В начале процесса положим

$$\tilde{N}_i^k = 0; \; \tilde{P}_i^k = 0$$

для всех $i = 1, 2, ..., m+1$, $k = 1, 2, ..., n$.

Шаг 0 (предварительный). Пометим вершину $(m+1)^k$, $k = 1, 2, ..., n$ меткой

$$r_{m+1}^k = (1, 0, 0); \; f_{m+1}^k = 0; \; l_{m+1}^k = 0. \qquad (8)$$

Положим для этой вершины $\tilde{N}_{m+1}^k = 1$. Переходим к шагу 1.

Шаг 1. Пусть $\tilde{J}^k$ – множество помеченных вершин в k-м графе. Для вершин $t^k \in J^k$ и j^k рассмотрим все возможные числа

$$l_j^k = l_t^k + \alpha_{t^k j^k} \le L^k.$$

Если такие числа существуют, то найдём среди них наименьшее. Положив $\tilde{l} = \min_{jk} l_j^k$, перейдём к шагу 2. Если таких чисел не существует, перейдём к шагу 3.

Шаг 2. Рассмотрим одну из помеченных вершин t^k. Берём одну из её меток, для которых $r_t^{1k} > \tilde{P}_t^k$. Пусть это будет метка

$$r_t^k = \left(r_t^{1k}, r_t^{2k}, r_t^{3k}\right); \; f_t^k; \; l_t^k.$$

Пометим вершину j^k, для которой $l_t^k + \alpha_{t^k j^k} = \tilde{l}$, меткой

$$r_j^k = \left(\tilde{N}_j^k + 1, t, r_t^{1k}\right); \; f_j^k = f_t^k \cup j^k; \quad l_j^k = l_t^k + \tilde{l} \qquad (9)$$

лишь в том случае, если среди всех построенных меток этой вершины не существует такой, что $f_t^k \cup j^k \subseteq f_j^k$.

Для вновь помеченной вершины $\tilde{N}_j^k$ увеличиваем на единицу. Продолжаем процедуру пометок до тех пор, пока это возможно. Затем переходим к шагу 1.

Шаг 3. Полагаем

$$\tilde{P}_i^k = \tilde{N}_i^k, \; i = 1, 2, ..., m.$$

Далее просматриваются метки у каждой из помеченных вершин t^k. Если существует такая метка

$$r_t^k = \left(r_t^{1k}, r_t^{2k}, r_t^{3k}\right); \; f_t^k; \; l_t^k,$$

что соответствующий ей путь не продолжен ни в одну из вершин ($l_j^k = l_t^k + \alpha_{t^k j^k} > L^k$), то помечаем вершину $i^k = (m+1)^k$ меткой

$$r_{m+1}^k = \left(\tilde{N}_{m+1}^k + 1, t, r_t^{1k}\right); \; f_{m+1}^k = f_t^k; \qquad (10)$$

$$l_{m+1}^{k} = L^{k} - \theta_{ks}$$

лишь в том случае, если среди всех уже построенных меток этой вершины не существует такой, что для её признака f_{m+1}^{k} справедливо соотношение $f_t^k \subseteq f_{m+1}^k$.

Увеличим $\tilde{N}_{m+1}^{k}$ на единицу. процедуру пометки указанной вершины продолжаем до тех пор, пока это возможно. Затем увеличиваем L^k на L_{ks}, которое определяется согласно (8), прибавим к s^k единицу и переходим к шагу 1. Если $L_{ks} = 0$, то переходим к шагу 4.

Шаг 4. Определим множество вершин $J_\mu \subseteq f_{m+1}^{k}$ в k-м графе ($k = 1, 2, ..., n$), неэквивалентное ранее рассмотренному:

$$\bigcup_{\mu} J_\mu = J\,;\ J_\mu \subseteq J\,;\ (m+1) \in J\,;\ J : \{1, 2, ..., m, m+1\}\,;$$

$$\mu = 1, 2, ..., \sigma\,;\ \sigma = 2^m - 1.$$

Если такое множество существует, то проверяем

$$\sum_{\substack{k_1=1 \\ k_1 \neq k}}^{n} \left(G_{k_1} - \sum_{s} \theta_{k_1 s} \right) \geq \geq \sum_{j \in J/J_\mu} \min_{k_1=1,...,n} \min_{i^k, j^k \in J/J_\mu} \left(\alpha_{i^k j^k} \right). \tag{11}$$

Если справедливо (11), то путь, проходящий в k-м графе через множество вершин J_μ, может входить в допустимое расписание. Переходим к шагу 5. В противном случае определим другое множество вершин J_μ^k. Если таких множеств не существует, то алгоритм построения допустимых путей в k-м графе заканчивает свою работу.

Шаг 5. Среди пометок вершин графа, признак которых $f_j^k = J_\mu^k$, выберем и запомним такую, для которой

$$l_{jq}^{k} = \min_{j^k} l_j^k\,;\ f_j^k = J_\mu^k\,; \tag{12}$$

Шаг 6. Восстанавливаем путь, заканчивающийся в вершине j_q^k и проходящий через множество вершин J_μ^k. Рассмотрим признаки

$$r_{j_q}^{k} = \left(r_{j_q}^{1k}, r_{j_q}^{2k}, r_{j_q}^{3k} \right) -$$

метки этой вершины. Перейдём к вершине

$$j_{q-1}^{k} = r_{j_q}^{2k}$$

и рассмотрим признаки

$$r_{j_{q-1}}^{k} = \left(r_{j_{q-1}}^{1k}, r_{j_{q-1}}^{2k}, r_{j_{q-1}}^{3k} \right) -$$

метки этой вершины, для которой

$$r_{j_{q-1}}^{1k} = r_{j_q}^{3k}.$$

Далее перейдём к вершине

$$j_{q-2}^{k} = r_{j_{q-1}}^{2k}$$

и так далее до тех пор, пока не перейдём к вершине $i^k = (m+1)^k$, для которой

$$r_{m+1}^{3k} = 0.$$

Последовательность вершин

$$\left\{ (m+1)^k, j_1^k, j_2^k, ..., j_q^k \right\}$$

и будет искомым путём; её вместе с признаками $l_{j_q}^k$ и $f_{j_q}^k$ – метки вершины j_q^k – сохраняем в памяти компьютера, а признаки $f_{j_q}^k$ и $l_{j_q}^k$ заносим в таблицу "перспективных" путей в k-м графе, которые могут входить в допустимое расписание. Переходим к шагу 4.

Алгоритм построения допустимых расписаний работы g машин ($g = 1, 2 ..., n$)заключается в дополнении каждого допустимого расписания работы ($g - 1$) машин всеми допустимыми расписаниями работы g-й машины, в проверке, может ли обеспечить построенное расписание выполнение производственной программы, и отсеве "неперспективных вариантов".

Каждое допустимое расписание работы g машин помечется меткой, состоящей из следующих ($g + 2$) признаков: первые g признаков метки указывают множество операций, выполняемых на каждой машине

$$f_\xi^k\ (f_\xi^{k_1} \cap f_\xi^{k_2} = \varnothing)\,;$$

($g + 1$)-й признак метки указывает множество операций, выполняемых на всех машинах

$$F_g = \bigcup_{k=1}^{g} f_\xi^k\,,$$

а ($g + 2$)-й – суммарное время работы всех машин,

$$M_g = \sum_{k=1}^{g} l_\xi^k\,.$$

Пусть построены расписания работы ($g - 1$) машин, которые сведены в табл. 1 (см. ниже). Кроме того, в табл. 2 записаны допустимые расписания работы g-й машины, которые помечаются меткой, состоящей из двух признаков: первый признак указывает множество операций, выполняемых на машине f_ξ^g, а второй – время работы машины l_ξ^g. Правило конструирования вариантов расписания работы g машин заключается в следующем: ($g-1$) признаков метки остаются без изменения, g-й признак метки полагается равным f_ξ^g, ($g+1$)-й признак –

$$F_g = F_{g-1} \cup f_\xi^g ,$$

а ($g+2$)-й –

$$M_g = M_{g-1} - l_\xi^g .$$

Расписание считается допустимым, если

$$F_{g-1} \cup f_\xi^g = \varnothing ;$$

$$\sum_{k=g+1}^{n} \left(G_k - \sum_{s=1}^{r} \theta_{ks} \right) \geq \qquad (13)$$

$$\geq \sum_{j \in J/F_g} \min_{k=(g+1),\ldots,n} \min_{i,j \in J/F_g} \left(\alpha_{i^k j^k} \right) .$$

Если окажется, что в табл. 1 или 2 есть расписание, для которого

$$F_g \subseteq F_{g-1},\ M_g \geq M_{g-1},$$

либо

$$F_g \subseteq f_\xi^g,\ l_\xi^g \leq M_g ,$$

то это расписание и записывается в табл. 3. После построения всех допустимых расписаний работы g машин табл. 3 играет роль табл. 1, а в табл. 2 записываются все допустимые расписания работы ($g+1$)-й машины.

Согласно этим правилам могут быть построены все перспективные расписания работы n машин. Среди расписаний, для которых $F_n = J$, выбирается такое, у которого

$$M_n = \min .$$

Это и будет график работы машин, обеспечивающий выполнение производственной программы с минимальными затратами машинного времени. Первые n признаков метки этого расписания указывают множество операций, выполняемых на различных машинах, по которым, обращаясь к буферной памяти компьютера, можно восстановить последовательность и времена выполнения этих операций на различных рабочих местах.

Пример построения оптимального расписания

На двух параллельных рабочих местах необходимо выполнить пять независимых операций, без разрывов во времени выполнения. Времена выполнения i-й операции на k-й машине (в часах) заданы матрицей A

$$A = \begin{vmatrix} 17 & 15 \\ 18 & 17 \\ 13 & 14 \\ 12 & 12 \\ 16 & 18 \end{vmatrix},$$

времена переналадок оборудования – матрицами B_1 и B_2

$$B_1 = \begin{vmatrix} 0 & 6 & 7 & 6 & 5 \\ 8 & 0 & 7 & 6 & 4 \\ 7 & 6 & 0 & 5 & 7 \\ 5 & 7 & 4 & 0 & 6 \\ 4 & 6 & 5 & 3 & 0 \end{vmatrix}; \quad B_2 = \begin{vmatrix} 0 & 8 & 7 & 6 & 5 \\ 4 & 0 & 5 & 5 & 4 \\ 5 & 6 & 0 & 7 & 4 \\ 8 & 9 & 7 & 0 & 7 \\ 5 & 6 & 7 & 6 & 0 \end{vmatrix},$$

времена пуско-наладочных работ, необходимые для подготовки рабочего места к работе в i-м режиме после планового простоя – векторами D_1 и D_2

$$D_1 = \begin{vmatrix} 4 \\ 5 \\ 5 \\ 6 \\ 7 \end{vmatrix}; \quad D_2 = \begin{vmatrix} 6 \\ 7 \\ 6 \\ 5 \\ 7 \end{vmatrix}.$$

Необходимо найти расписание работы машин, обеспечивающее выполнение производственной программы с наименьшими затратами машинного времени, если известно, что время работы первой машины не может превышать 54, а второй – 60 часов.

Допустимые расписания работы 1-й и 2-й машин сведены в табл. 1 и 2, а расписания работы двух машин, обеспечивающие выполнение производственной программы, – в табл. 3.

Таблица 1

№ п/п	f_ξ^1	l_ξ^1
1	4,5	37
2	1,4	38
3	3,5	39
4	1,5	41
5	2,3	42
6	5,2	45
7	2,1	46
8	3,4,5	53

Таблица 2

№ п/п	f_ξ^2	l_ξ^2
1	1,2	42
2	4,1,3	57
3	4,2,3	59
4	4,5,3	59

Таблица 3

№ п/п	f_ξ^1	f_ξ^2	F_2	M_2
1	2,5	4,1,3	1,2,3,4,5	102
2	1,5	4,2,3	-"-	100
3	2,1	4,5,3	-"-	105
4	3,4,5	1,2	-"-	95

Оптимальное расписание, кроме того, обеспечивает и выполнение производственной программы в кратчайшее время $\min_k \max l_\xi^k = 53$. Таким образом, задача решена.

Заключение

В статье рассмотрена задача формирования расписаний работ рабочих мест, обеспечивающего выполнение производственной программы с наименьшими затратами машинного времени при наличии ограничений на общую загрузку машин. Дана формально-математическая постановка задачи. Выполнен анализ свойств задачи. Предложен алгоритм её решения. Теоретические результаты проиллюстрированы примером формирования расписания.

Результаты работы могут применяться при проектировании и разработке программного обеспечения автоматизированных систем планирования и управления ресурсами в производстве сложных изделий.

Литература

1. *Сычёв В.А. Управление машиностроительным производством с помощью APS-систем / В.А. Сычёв // Вестник ЮРГТУ, 2012, №5. – С. 6 – 13.*

2. *Каплунов М.И. Календарное планирование для унифицированной модели объекта управления дискретного производства / М.И. Каплунов // Математичні моделі і системи, 2004, № 1. – С. 76 – 85.*

3. *Зак Ю.А. Построение оптимальных расписаний программных движений промышленных роботов / Ю.А. Зак // Управляющие машины и системы, 2014, №2. – С. 66 – 76.*

4. *Лазарев А.А. Теория расписаний. Исследование задач с отношениями предшествования и ресурсными ограничениями / А.А. Лазарев, Е.Р. Гафаров // Научное издание. Вычислительный Центр им. А.А. Дородницына РАН, 2007.– 80с.*

5. *Лазарев А.А. Теория расписаний. Задачи и алгоритмы. / А.А. Лазарев, Е.Р. Гафаров. – М.: МГУ им. М.В. Ломоносова, 2011. – 222 с.*

References

1. Sychev, V.A. (2012). *Engineering production control with the help of APS-systems. Bulletin of JuRGTU, No 5, 6-13*

2. Kaplunov, M.I. (2004). *Scheduling for a unified model of the control object discrete manufacturing. Mathematical models and systems, No 1, 75-86*

3. Zak, Yu.A., (2014). *Construction of optimal scheduling software movements of industrial robots.* Control Machines and Systems, No2, 66-76.

4. Lazarev, A.A., Gafarov, E.R. (2007). *Theory schedules. Research tasks with precedence relations and resource constraints.* Scientific edition. Dorodnicyn Computer Centre, RAN, 80.

5. Lazarev, A.A., Gafarov, E.R. (2011). *Theory schedules. The tasks and algorithms.* Moskow: Lomonosow MGU, 222.

Автор: ЛЫСЕНКО Дмитрий Эдуардович, к.т.н., доцент, докторант кафедры прикладной математики и информационных технологий Одесского национального политехнического университета.

МОДЕЛІ СКЛАДАЛЬНИХ РОБІТ В ОДИНИЧНОМУ Й ДРІБНОСЕРІЙНОМУ ВИРОБНИЦТВІ

Д.Е. Лисенко

Робота присвячена плануванню виробничих процесів у складальному виробництві. Розглянуто завдання формування розкладів робіт робочих місць, що забезпечує виконання виробничої програми з найменшими витратами машинного часу при наявності обмежень на загальне завантаження машин. Побудовано математичну модель завдання. Виконано аналіз властивостей запропонованої математичної моделі. Запропоновано алгоритм її вирішення. Теоретичні результати проілюстровані прикладом формування розкладу.

Ключові слова: *складальне виробництво, теорія розкладів, математичне моделювання*

MODEL OF ASSEMBLY WORK IN INDIVIDUAL AND SMALL-SCALE PRODUCTION

D.E. Lysenko

Work is devoted to the planning of production processes in assembly-line production. The task of forming a schedule of works jobs, providing the fulfillment of the production program with the lowest cost of working time under constraints on the total workload of machines. A mathematical model of the problem is build. The analysis of the properties of the proposed mathematical model is performed. An algorithm to solve this problem is offered. Theoretical results are illustrated by example of creating schedules.

Keywords: *assembly production, scheduling theory, mathematical modeling.*