УДК 681.513.5

МОДЕЛИРОВАНИЕ ТЕХНОЛОГИЧЕСКОГО ПРОЦЕССА С ИСПОЛЬЗОВАНИЕМ СЕТЕЙ ПЕТРИ

В. В. Шмелев, канд. техн. наук; *В. В. Мышко*
Военно-космическая академия им. А. Ф. Можайского, Санкт-Петербург, Россия

А. В. Евенко
Московский авиационный институт (национальный исследовательский университет), Москва, Россия

Рассматривается подход к моделированию технологических процессов в сложных технических системах. Процессы представляются в виде сетевой структуры (модели), построенной на принципах сетей Петри. Сетевая структура изображается в нотациях сетей Петри и формализуется с помощью теоретико-множественного подхода к изображению моделей. Показывается вариант математической структуры выбора оптимального управления на модели. Рассматриваются особенности применения метода динамического программирования для решения оптимизационной задачи для процесса, модель которого представлена сетевой структурой.

Ключевые слова: модель технологического процесса, сети Петри, математическая структура выбора, метод динамического программирования.

Автоматизированное управление технологическими процессами предполагает обязательное формирование пяти составляющих: цель и ограничения (ресурсы) выполнения процесса, альтернативы и критерии выбора при выполнении процесса и модель самого процесса. Важное значение имеет способ моделирования технологического процесса. Предлагается структурно-логический подход [1, 2] к синтезу, верификации, контролю и управлению моделями технологических процессов. Каждая операция процесса представляется в виде универсальной сетевой структуры, построенной на принципах сетей Петри обобщенной схемы операции технологического процесса. Такая универсальная сетевая структура операции является отличительной особенностью структурно-логического подхода. Моделирующие возможности подхода с учетом вводимых модификаций сетей Петри в достаточной степени удовлетворяют сложности логических взаимосвязей разнообразных технологических процессов.

Шмелев Валентин Валерьевич, докторант.
E-mail: valja1978@yandex.ru
Мышко Василий Васильевич, доцент кафедры технологии и средств комплексной обработки и передачи информации в АСУ.
E-mail: Myshko@yandex.ru
Евенко Александр Валерьевич, старший преподаватель учебного военного центра.
E-mail: evenko_av@mail.ru

Статья поступила в редакцию 31 мая 2016 г.

Математическая модель технологического процесса

Сетевая структура графически представляется в нотациях модифицированных сетей Петри [3], известных под названием G-сетей [4], а формализуется с помощью теоретико-множественного подхода следующим образом:

– $P = \{P_{\text{вн}}, P_{in}, P_{out}\} = \{p_i \mid i \in I_P\}$ — конечное непустое множество позиций сети Петри S, I_P — множество номеров позиций сети, $P_{\text{вн}}, P_{in}$, P_{out} — множества, соответственно, внутренних, входных и выходных позиций сети;

– $T = \{t_j \mid j \in I_T\}$ — конечное непустое множество переходов сети Петри S, I_T — множество номеров переходов сети;

– $F: P \times T \to N$ — входная функция инцидентности, описывающая кратность входной дуги от позиции p_i к переходу t_j сети S и ставящая в соответствие каждой паре $\langle p_i, t_j \rangle$, $i \in I_P$, $j \in I_T$ элемент множества целых неотрицательных чисел N;

– $B: P \times T \to Nb$ — входная функция инцидентности, описывающая сбрасывающую дугу от позиции p_i к переходу t_j сети S и ставящая в соответствие каждой паре $\langle p_i, t_j \rangle$, $i \in I_P$, $j \in I_T$, элемент бинарного множества $Nb = \{0,1\}$;

– $H^+: T \times P \to N$ — выходная функция инцидентности, описывающая кратность выходной «классической» дуги от перехода t_j в позицию p_i сети S и ставящая в соответствие каждой паре $\langle t_j, p_i \rangle$, $i \in I_P$, $j \in I_T$, элемент множества целых неотрицательных чисел N;

– $H^-: T \times P \to N$ — выходная функция инцидентности, описывающая кратность выходной извлекающей («неклассической») дуги от перехода t_j в позицию p_i сети и ставящая в соответствие каждой паре $\langle t_j, p_i \rangle$, $i \in I_P$, $j \in I_T$, элемент множества целых неотрицательных чисел N;

– $M: P \to N$ — функция разметки, которая каждому элементу $p_i \in P$ ставит в соответствие элемент множества целых неотрицательных чисел N, $M = \{\mathrm{M_{вн}}, \mathrm{M}_{in}, \mathrm{M}_{out}\}$.

Таким образом, модель отдельной операции можно описать следующим выражением:

$$S = \{P, T, F, B, H^+, H^-, M\}. \qquad (1)$$

При формальном описании технологического процесса необходимо дополнить выражение (1) логическими связями между операциями и ограничениями на траекторию развития процесса. Логические связи между операциями целесообразно описать с помощью функций инцидентности, а ограничения — с помощью отношений, уменьшающих множество допустимых альтернатив развития процесса. Модель технологического процесса $\Re$ может быть представлена следующим множеством:

$$\Re = \{S, \Im, Q\}, \qquad (2)$$

где $S = \left\{S_k \,\middle|\, k = \overline{1, \mathrm{card}(I_S)}\right\}$ — множество операций технологического процесса $\Re$, S_k — k-я операция, входящая в технологический процесс $\Re$, I_S — целочисленное положительное непустое ограниченное множество номеров операций, тогда $\mathrm{card}(I_S)$ — мощность множества операций;

$\Im = \left\{\Im_k \,\middle|\, k = \overline{1, \mathrm{card}(I_S)}\right\}$ — множество функций инцидентности технологического процесса $\Re$, описывающее логику процесса, $\Im_k$ — функция инцидентности операции S_k, где $\Im_k : P_{in}^{(l,k)} \times P_{out}^{(k,m)} \to N_\Im$ описывает «склеивание» выходных позиций операции S_l и входных позиций операции S_k, а также выходных позиций операции S_k и входных позиций операции S_m, $l, k, m \in I_S$. Перечень «склеиваемых» входных позиций из множества $P_{in}^{(l,k)}$ и выходных из множества $P_{out}^{(k,m)}$ определяет логику движения моделируемого технологического процесса;

$Q = \left\{Q_k \,\middle|\, k = \overline{1, \mathrm{card}(I_S)}\right\}$ — множество ограничений технологического процесса $\Re$, Q_k — множество отношений, ограничивающих выбор альтернативы развития k-й операции (например, запуска операции). Естественно, данное множество может быть и пустым при отсутствии альтернатив. В свою очередь, ограничение Q_k операции S_k также является множеством $Q_k = \left\{q_c^{(k)} \,\middle|\, c = \overline{1, \mathrm{card}(Q)}\right\}$, где $q_c^{(k)}$ — c-й вид ограничения операции S_k, c — порядковый номер ограничения, $\mathrm{card}(Q)$ — количество накладываемых ограничений.

Функционирование универсальной сетевой структуры операции может быть проверено, например, в среде CPN Tools [5, 6].

Математическая структура выбора

Технологический процесс подразумевает выполнение определенной последовательности действий, предусмотренных прикладной технологией проведения операций, позволяющей перевести процесс из некоторого начального состояния в требуемое. Множество вариантов развития процесса, позволяющих осуществить такой переход, образуют множество Δ_D допустимых альтернатив выполнения процесса. При этом на процесс накладывается ряд условий и ограничений. Для задания множества Δ_D может быть использовано отношение

$$\Delta_D : \Delta \times M_{\text{нач}\ out} \times M_{\text{кон}\ out} \times Q \to \Delta_D, \qquad (3)$$

где Δ — множество всех вариантов развития технологического процесса;

$M_{\text{нач } out}$ — множество начальных (исходных) состояний технологического процесса;

$M_{\text{кон } out}$ — множество требуемых состояний технологического процесса;

Q — множество отношений, ограничивающих выбор.

Для всего технологического процесса

$$M_{\text{нач } out} = \left\{ M_{\text{нач } k \text{ } out} \middle| k = \overline{1, \text{card}\left(I_S\right)} \right\},$$

$$M_{\text{кон } out} = \left\{ M_{\text{кон } k \text{ } out} \middle| k = \overline{1, \text{card}\left(I_S\right)} \right\}.$$

Последние выражения обозначают начальные и конечные разметки выходных позиций обобщенных схем всех операций технологического процесса. Данные множества могут иметь как единичную мощность, так и мощность более единицы. В первом случае краевые условия задаются точечно, во втором случае краевые условия на состояния технологического процесса задаются в виде интервала значений.

Множество Δ содержит все варианты выполнения технологического процесса, т. е. все возможные кортежи допустимых разметок входных позиций, обеспечивающих переход управляемого процесса из начального ($M_{\text{нач } out}$) в требуемое ($M_{\text{кон } out}$) состояние.

Множество отношений $Q = \left\{ Q_k \middle| k = \overline{1, \text{card}\left(I_S\right)} \right\}$, ограничивающих выбор среди элементов множества Δ, содержит ограничения различного рода на выполнение процесса [7].

Таким образом, математическая структура выбора может быть представлена в виде

$$\Delta^* = K\left(\mathfrak{R}, \Delta, Q, R\right), \qquad (4)$$

где K — некоторая функция, значением которой является оптимальный кортеж Δ^* разметок входных позиций обобщенных схем операций технологического процесса, а аргументами — следующие множества: $\mathfrak{R}$ — модель технологического процесса, Δ — множество вариантов кортежей разметок входных позиций обобщенных схем операций технологического процесса; Q — множество отношений, ограничивающих выбор и отражающих технологические, технические, краевые, ресурсные, пространственно-временные и другие требования, $R = \left\{ r_i, i = \overline{1, \text{card}\left(R\right)} \right\}$ — множество отношений предпочтения (показателей эффективности), определяющих выбор оптимального кортежа Δ^* разметок входных позиций обобщенных схем операций технологического процесса.

Метод динамического программирования

После формирования математической структуры выбора (4) необходимо рассмотреть непосредственно применение известных инструментов нахождения оптимального кортежа Δ^* разметок входных позиций обобщенных схем операций технологического процесса. Для решения задач управления в динамической интерпретации разработан достаточно мощный методологический аппарат анализа и синтеза. Из всего многообразия методов решения задач такого типа наиболее часто используется принцип оптимальности, положенный в основу метода так называемого динамического программирования [8]. Условия применения такого метода следующие.

Во-первых, желаемым условием является дискретность функционирования обобщенных схем операций технологического процесса и, следовательно, модели всего процесса в целом. Именно для задач оптимального управления с дискретным временем метод динамического программирования наиболее хорошо проработан. Для задач оптимального управления с непрерывным временем, описываемым дифференциальными управлениями, теория нахождения решений в последние годы активно разрабатывается. При этом функционирование технических систем зачастую удобнее рассматривать именно в дискретной постановке, приближаясь к непрерывной уменьшением периода дискретизации времени.

Во-вторых, необходимым условием применения метода динамического программирования являются ограниченность количества $\text{card}\left(\mathrm{M}_{out}\right)$ возможных состояний модели процесса и, кроме того, закрепление левого $\left(M_{\text{нач } out}\right)$ и правого $\left(M_{\text{кон } out}\right)$ концов фазовой траектории моделируемого процесса. Обоснованность выдвижения данного условия заключается в необходимости построения так называемой обратной процедуры динамического программирования.

В-третьих, необходимым условием является аддитивность показателя оптимальности. В рассматриваемой структуре выбора (4) результирующим показателем оптимальности является некоторая функция элементов r_i, $i = \overline{1, \mathrm{card}(R)}$ множества R. Известно [8], что принцип оптимальности является следствием аддитивности показателя оптимальности и не имеет места в случае неаддитивного показателя.

Многошаговый процесс управления

Рассмотрим управляемый процесс $\Re$, состояние M_{out} которого характеризуется разметками $M_{k\ out} \big| k = \overline{1, \mathrm{card}(I_S)}$ выходных позиций P_{out} обобщенных схем $S_k \big| k = \overline{1, \mathrm{card}(I_S)}$ операций процесса $\Re$. Для определенности примем, что смена состояния процесса происходит с получением отсчетов событий ОС. Предполагаем, что на каждом шаге на процесс оказывается управляющее воздействие, заключающееся в изменении разметки $M_{in}(\mathrm{OC})$ входных позиций P_{in} обобщенных схем. Таким образом, в каждый момент ОС состояние процесса характеризуется разметкой $M_{out}(\mathrm{OC})$, а управляющее воздействие — разметкой $M_{in}(\mathrm{OC})$. Так как нами используются только факты получения ОС, можно считать, что переменная ОС изменяется дискретно и принимает целочисленные значения 0, 1 и т. д., имеющие смысл порядковых номеров отсчетов. На выбор управления наложены ограничения Q, которые в достаточно общей форме можно представить в виде

$$q(\mathrm{OC}) \in Q, \quad \mathrm{OC} = 0, 1, \ldots \tag{5}$$

Под влиянием выбранного при получении очередного ОС управления (принятого решения) процесс переходит в новое состояние. Этот переход можно описать соотношением

$$M_{out}(\mathrm{OC}+1) = f\Big(M_{out}(\mathrm{OC}), M_{in}(\mathrm{OC})\Big). \tag{6}$$

Здесь $f\Big(M_{out}(\mathrm{OC}), M_{in}(\mathrm{OC})\Big)$ — функция, характеризующая динамику рассматриваемого процесса, определяемая моделью $\Re$ технологического процесса. Эта функция предполагается известной (заданной) и отвечает принятой математической модели рассматриваемого управляемого процесса.

Зададим еще начальное состояние системы

$$M_{out}(0) = M_{\text{нач}\ out}. \tag{7}$$

Таким образом, многошаговый процесс управления описывается соотношениями (5)—(7). Процедура расчета конкретного процесса сводится к следующему. Пусть с получением некоторого отсчета ОС состояние $M_{out}(\mathrm{OC})$ процесса известно. Тогда для определения состояния $M_{out}(\mathrm{OC}+1)$ необходимо выполнить две операции: 1) выбрать допустимое управление $M_{in}(\mathrm{OC})$, удовлетворяющее условию (5); 2) определить состояние $M_{out}(\mathrm{OC}+1)$ перед получением следующего ОС. Так как начальное состояние процесса известно, описанную процедуру можно выполнить последовательно для всех $\mathrm{OC} = 0, 1, \ldots$

Задача оптимального управления

Пусть задан некоторый критерий качества процесса управления (критерий оптимальности) вида

$$J = \sum_{\mathrm{OC}=0}^{N-1} R\Big(M_{out}(\mathrm{OC}), M_{in}(\mathrm{OC})\Big) + F\Big(M_{out}(N-1)\Big). \tag{8}$$

Выражение (8) — это способ представления элемента J структуры выбора (4). Здесь $R\Big(M_{out}(\mathrm{OC}), M_{in}(\mathrm{OC})\Big)$ и $F\Big(M_{out}(N-1)\Big)$ — заданные скалярные функции своих аргументов, $N-1$ — момент окончания процесса, $N > 0$. При этом функция R — показатель эффективности, определяющий выбор оптимального управления, а функция F характеризует оценку конечного состояния процесса или точность приведения в заданное состояние $M_{out}(N-1) = M_{\text{кон}\ out}$.

Задача оптимального управления формулируется как задача определения допустимых управлений (кортежа)

$$\Delta^* = \langle M_{in}(0),\ M_{in}(1),\ ...,\ M_{in}(N-1)\rangle,$$

удовлетворяющих ограничениям (5), и соответствующей траектории, т. е. последовательности (так же кортежа)

$$\Big\langle M_{out}(0) = M_{\text{нач}\ out},\ M_{out}(1),\ ...,\ M_{out}(N-1) = M_{\text{кон}\ out}\Big\rangle,$$

которые в совокупности доставляют минимальное (максимальное) значение критерию (8) для процесса (6) и (7). Можно записать

$$\Delta^* = \langle M_{in}(0),\ M_{in}(1),\ ...,\ M_{in}(N-1)\rangle = = \arg \min_{\langle M_{out}(0)=M_{\text{нач}\ out},\ M_{out}(1),\ ...,M_{out}(N-1)=M_{\text{кон}\ out}\rangle}(\max) J.$$

Решить рассматриваемую задачу возможно элементарным подходом или простым перебором. Однако этот подход требует значительных вычислительных затрат. Принципиально иной подход к поставленной проблеме дает метод динамического программирования.

Сформулированный Р. Беллманом принцип оптимальности гласит [8]: отрезок оптимального процесса от любой его точки до конца сам является оптимальным процессом с началом в данной точке.

Метод динамического программирования

Обозначим как $J^*_{\min}\left(M_{out}\left(\text{ОС}^*\right), \text{ОС}^*\right)$ минимальное значение критерия качества J^* для оптимального процесса, начинающегося при получении ОС^* в состоянии $M_{out}\left(\text{ОС}^*\right)$. Здесь и далее в качестве экстремального значения будет рассматриваться минимальное значение функции критерия качества. Этот процесс можно представить состоящим из двух участков: первого шага, на котором выбирается управление $M_{in}\left(\text{ОС}^*\right)$, и остальной части (от момента $\left(\text{ОС}^*+1\right)$ до конца процесса). Вклад в показатель качества первого участка равен $R\left(M_{out}\left(\text{ОС}^*\right), M_{in}\left(\text{ОС}^*\right)\right)$, а вклад второго участка можно, согласно принципу оптимальности, выразить через введенную функцию $J^*_{\min}$ в виде $J^*_{\min}\left(M_{out}\left(\text{ОС}^*+1\right), \text{ОС}^*+1\right)$. Учитывая, что управление на первом участке должно выбираться из условия минимизации критерия J^* при ограничениях (5), получим равенство

$$J^*_{\min}\left(M_{out}\left(\text{ОС}^*\right), \text{ОС}^*\right) = = \min_{M_{in}\left(\text{ОС}^*\right)\in \mathrm{M}_{in}} \left[\begin{array}{l} R\left(M_{out}\left(\text{ОС}^*\right), M_{in}\left(\text{ОС}^*\right)\right) + \\ + J^*_{\min}\left(M_{out}\left(\text{ОС}^*+1\right), \text{ОС}^*+1\right) \end{array}\right].$$

Здесь M_{in} — множество всех возможных состояний разметок входных позиций фрагмента сети, допустимых с учетом (5).

Подставляя в полученное соотношение равенство (6), получим основное соотношение метода динамического программирования:

$$J_{\min}\left(M_{out}(\text{ОС}), \text{ОС}\right) = = \min_{M_{in}(\text{ОС})\in \mathrm{M}_{in}} \left[\begin{array}{c} R\left(M_{out}(\text{ОС}), M_{in}(\text{ОС})\right) + \\ + J^*_{\min}\left(f\left(\begin{array}{c} M_{out}(\text{ОС}), \\ M_{in}(\text{ОС}) \end{array}\right), \text{ОС}+1\right) \end{array}\right]. \quad (9)$$

Для оптимального процесса, начинающегося при получении последнего $\text{ОС} = N-1$, критерий оптимальности сводится к одному слагаемому. Поэтому имеем

$$J_{\min}\left(M_{out}(N-1), N-1\right) = F\left(M_{out}(N-1)\right). \quad (10)$$

Соотношение (9) и условие (10), играющее роль начального условия, дают возможность последовательно определить функции $J_{\min}\left(M_{out}(\text{ОС}), \text{ОС}\right)$ при $\text{ОС} = N-1, ..., 0$, а также рассчитать оптимальное управление и оптимальные траектории. Это достигается при последовательной реализации обратной и прямой процедур динамического программирования.

Обратная процедура

При вычислении минимума (максимума) функции по некоторому аргументу определяются две величины: минимальное (максимальное) значение функции и значение аргумента, при котором этот минимум (максимум) достигается. Последнее значение, которое может быть и не единственным, обозначается символом arg min для варианта поиска минимума функции и $\arg\max$ для варианта поиска максимума функции.

Положим в (9) $\mathrm{OC} = N-1$ и воспользуемся условием (10). Получим

$$J_{\min}\left(M_{out}(N-1), N-1\right) = = \min_{M_{in}(N-1)\in \mathrm{M}_{in}} \begin{bmatrix} R\left(M_{out}(\mathrm{OC}), M_{in}(\mathrm{OC})\right) + \\ +F\left(M_{out}(\mathrm{OC})\right) \end{bmatrix}.$$

Вычисляя этот минимум, найдем функцию $J_{\min}\left(M_{out}(N-1), N-1\right)$ и значение $M_{in}(\mathrm{OC})$, доставляющее данный минимум:

$$M_{in}(\mathrm{OC}) = \upsilon_{N-1}\left(M_{out}(\mathrm{OC})\right) = = \arg\min_{M_{in}(N-1)\in \mathrm{M}_{in}} \begin{bmatrix} R\left(M_{out}(\mathrm{OC}), M_{in}(\mathrm{OC})\right) + \\ +F\left(M_{out}(\mathrm{OC})\right) \end{bmatrix}.$$

Запись $\upsilon_{N-1}\left(M_{out}(\mathrm{OC})\right)$ означает, что значение $M_{in}(\mathrm{OC})$ зависит от $M_{out}(N-1)$ как от параметра. Определив $J_{\min}\left(M_{out}(N-1), N-1\right)$ и полагая $\mathrm{OC} = N-2$, найдем из (9) функцию $J_{\min}\left(M_{out}(N-2), N-2\right)$ и соответствующее значение аргумента $M_{in}(\mathrm{OC}) = \upsilon_{N-2}\left(M_{out}(\mathrm{OC})\right)$.

Продолжая этот процесс в сторону уменьшения ОС, получим из (9) последовательно функции $J_{\min}\left(M_{out}(\mathrm{OC}), \mathrm{OC}\right)$ и

$$\upsilon_{\mathrm{OC}}\left(M_{out}(\mathrm{OC})\right) = = \arg\min_{M_{in}(\mathrm{OC})\in \mathrm{M}_{in}} \begin{bmatrix} R\left(M_{out}(\mathrm{OC}), M_{in}(\mathrm{OC})\right) + \\ +J_{\min}\left(M_{out}(\mathrm{OC}), \mathrm{OC}\right) + 1 \end{bmatrix} \qquad (11)$$

при $\mathrm{OC} = N-1,\ N-2,\ \ldots,\ 1,\ 0$. Функция $\upsilon_{\mathrm{OC}} M_{out}(\mathrm{OC})$ определяет оптимальное управление при получении ОС при условии, что процесс находится в состоянии $M_{out}(\mathrm{OC})$.

Таким образом, обратная процедура состоит в построении функций $J_{\min}\left(M_{out}(\mathrm{OC}), \mathrm{OC}\right)$ и $\upsilon_{\mathrm{OC}}\left(M_{out}(\mathrm{OC})\right)$ для всех $M_{out}(\mathrm{OC})$ и $\mathrm{OC} = 0,\ 1,\ \ldots,\ N-1$. Это построение в наилучшем случае может быть выполнено аналитически, но, как правило, является трудоемкой вычислительной процедурой.

Прямая процедура

Воспользуемся результатами обратной процедуры для решения исходной задачи, т. е. для построения оптимального управления и оптимальной траектории при заданном начальном условии (7).

Полагая в (11) $\mathrm{OC} = 0$ и $M_{out}(0) = M_{\text{нач}\ out}$, найдем управление при получении первого ОС: $M_{in}(0) = \upsilon_0\left(M_{\text{нач}\ out}\right)$. Далее из соотношения (6) определим состояние $M_{out}(1) = = f\left(M_{\text{нач}\ out}, M_{in}(0)\right)$. Продолжая этот процесс, найдем $M_{in}(1) = \upsilon_1\left(M_{out}(1)\right)$, $M_{out}(2)$, $M_{in}(2)$ и т. д. Таким образом, имеем

$$\begin{gathered} M_{in}(\mathrm{OC}) = \upsilon_{\mathrm{OC}}\left(M_{out}(\mathrm{OC})\right), \\ M_{out}(\mathrm{OC}+1) = f\left(M_{out}(\mathrm{OC}), M_{in}(\mathrm{OC})\right), \\ M_{out}(0) = M_{\text{нач}\ out}, \\ \mathrm{OC} = 0,\ 1,\ \ldots,\ N-1. \end{gathered} \qquad (12)$$

Соотношения (12) определяют прямую процедуру и позволяют полностью рассчитать оптимальное управление и оптимальную траекторию. Минимальное (максимальное) значение критерия оптимальности, отвечающее этой траектории, есть $J = J_{\min}\left(M_{\text{нач}\ out}, 0\right)$.

Таким образом, обратная и прямая процедуры метода динамического программирования в совокупности дают способ решения поставленной задачи оптимального управления технологическим процессом, модель которого представлена совокупностью обобщенных схем операций.

Вычислительная реализация метода динамического программирования сталкивается с большими трудностями. Эти трудности Р. Беллман назвал проклятием размерности. Для преодоления этих трудностей могут быть предложены подходы [9], позволяющие сократить объем вычислений и потребности в памяти при построении оптимального управления. При этом, однако, приходится либо существенно пожертвовать точностью вычислений, либо отказаться от построения управления, оптимального в глобальном смысле, и ограничиться нахождением управлений и траекторий, оптимальных в локальном смысле, т. е. по отношению к малым (локальным) вариациям этих траекторий.

Заключение

Областью применения разработанного структурно-логического подхода и созданной на его основе модели технологического процесса могут быть информационные системы поддержки принятия решений в сложных технических системах. Возможность решения оптимизационных задач с помощью предлагаемой модели может значительно повысить эффективность таких систем.

В дальнейшем представляется важным рассмотрение особенностей применения методов многокритериальной оптимизации технологического процесса, модель которого создана с помощью структурно-логического подхода.

ЛИТЕРАТУРА

1. **Шмелев В. В.** Модели технологических процессов функционирования космических средств // Авиакосмическое приборостроение. 2015. № 4. С. 78—93.
2. **Шмелев В. В., Мануйлов Ю. С.** Применение модифицированных сетей Петри к моделированию процесса послеполетного анализа телеметрической информации // Электронный журнал «Труды МАИ». 2015. № 6(84). http://www.mai.ru/science/trudy/published.php?ID=63140
3. **Котов В. Е.** Сети Петри. — Л.: Наука, 1984.
4. **Охтилев М. Ю.** Основы теории автоматизированного анализа измерительной информации в реальном времени. Синтез системы анализа. Монография. — СПб.: ВКА им. А. Ф. Можайского, 1999.
5. Интернет ресурс: http://cpntools.org.
6. **Kristensen L. M., Christensen S., Jensen K.** // Int. J. Softw. Tools Technol. Transf. (1998) 2. P. 98—132.
7. **Шмелев В. В., Мануйлов Ю. С., Рахимов Р. Р., Богданов А. В.** Формализация технологического процесса на основе сетевой модели // Журнал «Научное обозрение». 2015. № 19. С. 156—161.
8. **Черноусько Ф. Л.** Динамическое программирование // Соросовский образовательный журнал. 1998. № 2. С. 139—144.
9. **Мануйлов Ю. С., Калинин В. Н., Гончаревский В. С. и др.** Управление космическими аппаратами и средствами наземного комплекса управления. — СПб.: ВКА им. А. Ф. Можайского, 2010.

MODELING PROCESS USING PETRI NETS

V. V. Shmelev, V. V. Myshko
Military Space Academy named A. F. Mozhajskij, St.-Petersburg, Russia

A. V. Evenko
Moscow Aviation Institute (National Research University), Moscow, Russia

An approach to modeling of processes in complex technical systems. A process is in the form of a network structure (model), built on the principles of Petri nets. The network structure is depicted in the notations of Petri nets and formalized using set-theoretic approach to image models. It shows a variant of the mathematical structure of the selection of the optimal control model. The features of the method of dynamic programming for solving the optimization problem for the process, which is represented by a network model structure.

Keywords: model of the process, Petri nets, mathematical structure selection, the dynamic programming method.

Bibliography — 9 references. *Received May 31, 2016*