УДК 681.513.5

Решение оптимизационной задачи на сетевой модели технологического процесса

Шмелев В.В.

Военно-космическая академия имени А.Ф. Можайского,

Ждановская ул., 13, Санкт-Петербург, 197198, Россия

e-mail: valja1978@yandex.ru

Аннотация

Рассматривается структурно-логический подход к синтезу, верификации, контролю и управлению моделями технологических процессов в сложных технических системах. Процессы представляются в виде сетевой структуры (модели), построенной на принципах сетей Петри. Моделирующие возможности инструмента сетей Петри с учетом вводимых модификаций в достаточной степени соответствуют разнообразию и сложности технологических процессов. Сетевая структура рассматривается в нотациях сетей Петри и с помощью теоретико-множественного подхода к изображению моделей. Показывается вариант математической структуры выбора оптимального управления на моделируемом процессе. 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.

Введение

Практическая реализация автоматизированного управления технологическими процессами функционирования и обработки информации сложных технических систем обеспечивается формированием следующих составляющих.

1. Цели. Целевой анализ начинается с формулировки глобальной цели управления технологическим процессом. Глобальная цель конкретизируется путем указания подчиненных ей главных целей. Дальнейшая конкретизация цели осуществляется с помощью формулирования задач, общих и частных.

2. Ограничения (ресурсы). Автоматизированное управление реальными техническими процессами должно осуществляться с учетом ограничений различного рода. Тем самым обеспечивается максимальная адаптация модели технологического процесса, в общем-то, именно с моделью осуществляется работа при автоматизированном управлении, событиям и состояниям реальной технической системы.

3. Альтернативы. При автоматизированном управлении должны быть выявлены (сконструированы) допустимые с учетом введенных ограничений альтернативы и, в дальнейшем, выделена из них наилучшая с известной точки зрения. Каждой альтернативе сопоставляется вариант управляющих воздействий на процесс в местах ветвления его фазовой траектории.

4. Критерии. Критерий – это правило, по которому осуществляется выбор или сравнение альтернатив. В качестве критерия выбора выдвигаются условия принадлежности альтернативы к множеству, обладающему определенными свойствами,

или достижения при этой альтернативе экстремума по некоторому показателю эффективности управляемого технологического процесса.

5. Модели. Исследование альтернатив производится на моделях. Эффективность управленческих решений обеспечивается достаточной адекватностью используемой модели технологического процесса.

Все рассмотренные элементы являются важными. Однако, представляется особенно значимой задача выбора адекватного способа моделирования, т.к. от него зависит, вообще, соответствие наблюдаемого состояния процесса реальному его состоянию и, следовательно, зависит эффективность принимаемых решений по управлению, и, в конечном счете, эффективность самого процесса.

В работе [1] предлагается подход к моделированию технологических процессов, в полной мере отвечающий особенностям технических процессов в такой сложной предметной области, как функционирование космических средств. Сетевая модель технологического процесса, созданная в соответствии со структурно-логическим подходом в достаточной степени адекватно способна представлять и процессы функционирования космических средств, и процессы обработки их информации. Это показывается в работе [2].

Однако, в указанных работах не рассмотрен вопрос формирования математической структуры выбора оптимального решения по управлению процессом, модель которого представлена предложенной в работе [1] сетевой структурой. Кроме того, актуально рассмотреть особенности применения метода динамического программирования, как одного из наиболее мощных и широко известных математических методов современной теории управления. В настоящей статье устраняется указанный пробел.

1. Математическая модель технологического процесса

Рассмотрим формальное описание технологического процесса. Напомним, что структурно-логический подход предполагает использование в качестве модели управляемого процесса модифицированную сеть Петри [3], известную под названием G-сети [4].

В соответствии с этим подходом структура технологического процесса представляется в виде совокупности неделимых операций с наложенными связями. Каждая операция представляет собой неизменную (универсальную) G-сеть. Связи между операциями являются отображением всех ограничений технологического процесса (технологических, краевых и др.). Реализуются связи с помощью инструмента сетей Петри. Таким образом, с помощью связей моделируется логика технологического процесса.

На первом этапе формального описания модели технологического процесса необходимо рассмотреть модель технологической операции, которая представляет собой атомарные, неделимые элементы, «кирпичики», из которых строится технологический процесс. На втором этапе в модель операции будет добавлено описание логики технологического процесса в виде описания связей между операциями.

Первый этап. Модель технологической операции удобно представить в графическом виде модифицированной сети Петри (рис. 1).

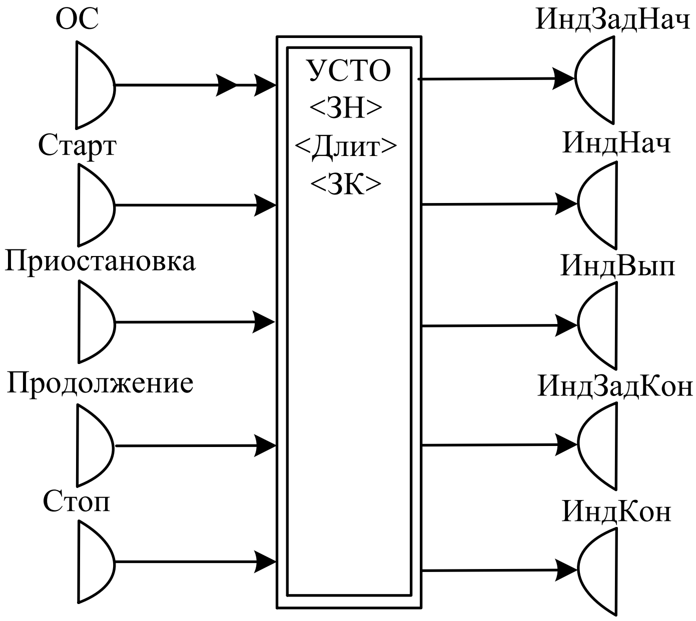

Рисунок 1 – Обобщенная схема операции

По причине универсальности схемы технологической операции, совокупность элементов сети Петри, за исключением входных и выходных позиций, может быть заменена специальным переходом-процедурой, который и является, по сути, универсальной схемой технологической операции (УСТО). Раскрывать внутреннюю структуру схемы в данном материале нет необходимости. Структура УСТО с описанием функционирования схемы представлена в работе [1].

УСТО обозначена прямоугольником с двойными линиями. В левой части приведены входные позиции, формирующие траекторию развития данной технологической операции и, следовательно, всего процесса в целом. В правой части приведены все выходные (индикаторные) позиции, отражающие траекторию развития данной технологической операции и, следовательно, всего процесса в целом. Внутри прямоугольника приведены значения длительностей задержки начала выполнения (<ЗН>), задержки окончания выполнения (<ЗК>) и непосредственно выполнения операции (<Длит>). Переход-процедура является удобным средством компактного отображения процессов. Вместо громоздкого объема внутренних позиций и переходов схемы модели ТП (смотри схему сети Петри в работе [1]) используется упрощенное изображение операции.

В позиции «Старт», «Стоп», «Приостановка» и «Продолжение» поступают управляющие сигналы. Получение фишек данными позициями приводит к началу выполнения операции («Старт»), к окончанию выполнения («Стоп»), к временной приостановке («Приостановка») и продолжению выполнения («Продолжение»). Данные позиции, совместно с выходными, обеспечивают выполнение требования управляемости модели операции. Позиция «ОС» (отсчет событий) содержит информацию о наступлении очередного элементарного события, используемого в качестве «счетчика» степени выполнения операции. В качестве элементарных событий могут использоваться как временные метки, так и события невременного характера, например, результаты телеизмерений технической системы.

Позиции «ИндЗадНач» (индикация значения задержки начала выполнения операции), «ИндНач» (индикация непосредственного начала выполнения операции), «ИндВып» (индикация значения степени выполнения операции), «ИндЗадКон» (индикация значения задержки окончания выполнения операции) и «ИндКон» (индикация факта окончания выполнения операции) являются индикаторными и обеспечивают наблюдаемость модели операции. Индикаторные позиции «ИндЗадНач», «ИндВып» и «ИндЗадКон» являются счетными, т.к. содержат количество отсчетов событий, полученных за интервал отсчета, соответственно, задержки начала выполнения, непосредственного выполнения и задержки окончания выполнения операции. Позиции «ИндНач» и «ИндВып» являются бинарными и «сигнализируют» о фактах, соответственно, начала и окончания выполнения операции.

Функционирование УСТО может быть проверено, например, в среде CPN Tools [5, 6].

С позиций теоретико-множественного подхода к описанию математических моделей можно описать модель операции как сеть Петри следующими множествами:

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

– $T = \{t_j | j \in I_T\}$ – 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 $< p_i, t_j >$, $i \in I_P$, $j \in I_T$ 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 $< p_i, t_j >$, $i \in I_P$, $j \in I_T$ 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 $< t_j, p_i >$, $i \in I_P$, $j \in I_T$ 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 $< t_j, p_i >$, $i \in I_P$, $j \in I_T$ 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.

Таким образом, модель отдельной операции можно описать следующим выражением:

$$S = \{P, T, F, B, H^+, H^-, M\} \tag{1}$$

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. В соответствии с выбранными способами представления логики и ограничений процесса модель технологического процесса $\Re$ может быть представлена следующим множеством:

$$\Re = \left\{ S, \Im, Q \right\}, \text{ где} \qquad (2)$$

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

– $\Im = \left\{ \Im_k \middle| k = \overline{1, card\left(I_S\right)} \right\}$ – 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 $S_m, m, l \in I_S$. Перечень «склеиваемых» входных позиций из множества $P_{in}^{(l,k)}$={«ОС», «Старт», «Стоп», «Приостановка», «Продолжение»} и выходных из множества $P_{out}^{(k,m)}$={«ИндЗадНач», «ИндНач», «ИндВып», «ИндЗадКон», «ИндКон»} определяет логику движения моделируемого технологического процесса. С учетом ограниченности количества входных и выходных позиций (мощности множеств $P_{in}^{(l,k)}$ и $P_{out}^{(k,m)}$) можно утверждать, что $N_{\Im}$ является множеством натуральных чисел, ограниченным сверху числом 5 включительно;

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

2. Математическая структура выбора оптимального варианта выполнения технологического процесса

Технологический процесс подразумевает выполнение определённой 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, речь о которых шла выше, обусловленных особенностями структурного и логического построения процесса, спецификой цели выполнения процесса, а также возможным влиянием факторов неопределенности среды. Для задания множества Δ_D может быть использовано отношение:

$$\Delta_D : \Delta \times \mathrm{M}_{\text{нач } out} \times \mathrm{M}_{\text{кон } out} \times Q \rightarrow \Delta_D, \text{ где} \quad (3)$$

– Δ – множество всех вариантов развития технологического процесса;

– $\mathrm{M}_{\text{нач } out}$ – множество начальных (исходных) состояний технологического процесса;

– $\mathrm{M}_{кон\ out}$ – множество требуемых состояний технологического процесса;

– Q – множество отношений, ограничивающих выбор.

Требуется дополнительное пояснение структуры элементов выражения (3).

В теории сетей Петри одним из способов описания смены состояний сети является траектория смен разметок сети [3], на основе, например, дерева достижимости. Некоторый элемент множества M можно использовать как отражение текущего состояния технологической операции. Обозначим через M_t некоторую разметку сети Петри или некоторое состояние операции, тогда $\mathrm{M} = \left\{ M_t \middle| t = \overline{1, card(\mathrm{M})} \right\}$ будет являться множеством всех возможных состояний разметок сети или множеством всех возможных состояний операции. Так как множество P состоит из трех непересекающихся подмножеств $P_{вн}$, P_{in} и P_{out}, то правомерным будет записать

$$\mathrm{M} : \mathrm{M}_{вн} \times \mathrm{M}_{in} \times \mathrm{M}_{out},$$

причем $card(\mathrm{M}) = card(\mathrm{M}_{вн}) \cdot card(\mathrm{M}_{in}) \cdot card(\mathrm{M}_{out})$, где $\mathrm{M}_{вн}$, M_{in} и M_{out} – множества возможных разметок фрагментов УСТО, состоящих из, соответственно, только внутренних, входных и выходных позиций $P_{вн}$, P_{out} и P_{in}, т. е. $\mathrm{M}_{вн} : P_{вн} \to N$ – функция разметки, которая каждому элементу $p_i \in P_{вн}$ ставит в соответствие элемент множества целых неотрицательных чисел N, $\mathrm{M}_{in} : P_{in} \to N$ – функция разметки, которая каждому элементу $p_i \in P_{in}$ ставит в соответствие элемент множества целых неотрицательных чисел N и $\mathrm{M}_{out} : P_{out} \to N$ – функция разметки, которая каждому элементу $p_i \in P_{out}$ ставит в соответствие элемент множества целых неотрицательных чисел N.

Из всей совокупности множества позиций $P=\left\{p_i \middle| i \in I_P\right\}=\left\{P_{вн}, P_{in}, P_{out}\right\}$ необходимо выделить множество P_{out}, т.к. именно это множество отражает состояние операции, и по смене разметок только этих позиций будет приниматься решение о наблюдаемом (текущем) состоянии моделируемой операции. Среди элементов множества M_{out} должны находиться и некоторые элементы $M_{нач\ out}$ и $M_{кон\ out}$, обозначающие, соответственно, начальное и конечное (требуемое) состояние технологической операции. Данные разметки должны входить, как минимум, в один кортеж разметок сети. При наличии нескольких подобных кортежей, содержащих разметки $M_{нач\ out}$ и $M_{кон\ out}$, операция должна допускать альтернативы в своем выполнении.

Для всего технологического процесса:

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

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

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.

Изменение состояния технологической операции произойдет после подачи управляющих сигналов на входные позиции P_{in} или формирования некоторой разметки $M_{t\ in} \in \mathrm{M}_{in}$. Вследствие срабатывания разрешенных переходов сети [3] будет сформирована новая разметка сети $M_{t'} = M_{t'\,вн} \bigcup M_{t'\,in} \bigcup M_{t'\,out}$ и технологическая операция перейдет в новое состояние, т.е. операция будет выполняться. Таким образом, элементарный кортеж $\langle M_t, M_{t'} \rangle$ описывает минимальный фрагмент траектории выполнения или развития технологической операции. Однако, в выражении (3) для составления множества Δ целесообразно использовать только «управляющую» часть кортежа разметок $M_{t'} = M_{t'\,вн} \bigcup M_{t'\,in} \bigcup M_{t'\,out}$, а именно кортежи множеств разметок $\mathrm{M}_{in} : P_{in} \to N$ входных позиций P_{in} по причине выбора именно среди них оптимального управления моделируемым процессом.

Таким образом, множество Δ содержит все варианты выполнения технологического процесса, т.е. все возможные кортежи допустимых разметок входных позиций, обеспечивающих переход управляемого процесса из начального $M_{нач\ out}$ в требуемое состояние $M_{кон\ out}$. В рамках моделей (1) и (2) такое множество будет содержать возможные кортежи разметок входных позиций обобщенной схемы операции (рис. 1) применительно к каждому элементу множества $S = \left\{ S_k \middle| k = \overline{1, card\left(I_S\right)} \right\}$. В итоге можно записать:

$$\Delta = \left\{ \Delta_t = \left\langle M_{нач\ in\ t}, \ldots, M_{кон\ in\ t} \right\rangle_t, \mathrm{t} = \overline{1, card\left(\Delta\right)} \right\}, \text{ причем}$$

$$\forall \mathrm{t} : \begin{pmatrix} M_{нач\ in\ t}, M_{кон\ in\ t} \in \mathrm{M}_{in} \\ M_{нач\ in\ t} : M_{нач\ in\ t} \times \Re \Rightarrow M_{нач\ out} \\ M_{кон\ in\ t} : M_{кон\ in\ t} \times \Re \Rightarrow M_{кон\ out} \end{pmatrix}.$$

Множество отношений $Q = \left\{ Q_k \middle| k = \overline{1, card\left(I_S\right)} \right\}$, ограничивающих выбор среди элементов множества Δ, содержит ограничения различного рода на выполнение процесса. Рассмотрим порядок использования ограничений процесса в модели процесса (2).

Придерживаясь концепции использования сетевой структуры для моделирования технологической операции, в рамках модели (1) разрешение для срабатывания перехода t_j задается неравенством:

$$m(p_i) \geq \left(f\left(< p_i, t_j >\right) + b\left(< p_i, t_j >\right)\right), \tag{4}$$

где $m(p_i)$ – значение функции разметки $M : P \to N$ для позиции $p_i \in P$, $i \in I_P$; $f\left(< p_i, t_j >\right)$ – значение функции инцидентности $F : P \times T \to N$ между элементами $p_i \in P$ и $t_j \in T$, $i \in I_P$, $j \in I_T$ или кратность дуг от позиции p_i к переходу t_j; $b\left(< p_i, t_j >\right)$ – значение функции инцидентности $B : P \times T \to N$ между элементами $p_i \in P$ и $t_j \in T$, $i \in I_P$, $j \in I_T$, или значение функции $b\left(< p_i, t_j >\right) = 1$ при наличии сбрасывающей дуги от позиции p_i к переходу t_j.

Вообще, существование задачи выбора обеспечивается наличием нескольких (более одного) вариантов выполнения технологического процесса. В случае использования инструмента сетей Петри для моделирования процесса существование нескольких вариантов выполнения процесса интерпретируется как конфликтная ситуация. Простейшим вариантом конфликтной ситуации является следующая модификация выражения (4):

$$\left(m(p_i) \geq \left(f\left(< p_i, t_j >\right) + b\left(< p_i, t_j >\right)\right)\right) \& \left(m(p_i) \geq \left(f\left(< p_i, t_k >\right) + b\left(< p_i, t_k >\right)\right)\right),$$

$$j \neq k; i \in I_P; j, k \in I_T.$$

Последнее выражение означает, что маркировка позиции p_i одновременно разрешает срабатывание переходов t_j и t_k.

Наиболее общим вариантом конфликтной ситуации (вторым вариантом) является следующая модификация выражения (4):

$$\Big(m(p_i) \geq \big(f(<p_i,t_j>) + b(<p_i,t_j>)\big)\Big) \& \Big(m(p_k) \geq \big(f(<p_k,t_l>) + b(<p_k,t_l>)\big)\Big),$$

$$i \neq k;\ \ j \neq l;\ \ i,k \in I_P; j,l \in I_T\ .$$

Выражение формализует маркировку сети, при которой одновременно разрешены для срабатывания более одного перехода. Рассмотренные выражения могут быть распространены на случай конфликта более двух переходов и позиций соответствующим увеличением количества аргументов в выражениях. Таким образом, показана математическая интерпретация выбора на классических сетях Петри и возможные варианты разрешения названного конфликта.

Необходимо интерполировать неравенство (4) на модель операции (выражение (1)), затем технологического процесса в целом (выражение (2)). Разрешение на изменение состояния операции:

$$m(p_i) \geq \big(f(<p_i,t_j>) + b(<p_i,t_j>)\big) \big| p_i \in P_{in} \Rightarrow S_k(1) \rightarrow S_k(2), \qquad (5)$$

где $S_k(1)$ и $S_k(2)$ – текущее и следующее состояния k-ой операции технологического процесса $\Re$, $k \in I_S$, $m(p_i)$ – значение функции разметки $M_{in}: P_{in} \rightarrow N$ для позиции $p_i \in P_{in}$, $i \in I_P$; $f(<p_i,t_j>)$ – значение функции инцидентности $F: P_{in} \times T \rightarrow N$ между элементами $p_i \in P_{in}$ и $t_j \in T$, $i \in I_P$, $j \in I_T$ или кратность дуг от позиции p_i к переходу t_j; $b\left(< p_i, t_j >\right)$ – значение функции инцидентности $B: P_{in} \times T \to N$ между элементами $p_i \in P_{in}$ и $t_j \in T$, $i \in I_P$, $j \in I_T$.

Выражение (5) интерпретируется следующим образом. Операция S_k сменит свое состояние из $S_k(1)$ в $S_k(2)$ при условии наличия хотя бы в одной из входных позиций хотя бы одной управляющей фишки.

Множество отношений $Q = \left\{ Q_k \middle| k = \overline{1, card(I_S)} \right\}$ содержит дополнительные множители, которые должны входить в (5) следующим образом:

$$Q_k \cdot m(p_i) \geq \left(f\left(< p_i, t_j >\right) + b\left(< p_i, t_j >\right)\right) \big| p_i \in P_{in} \Rightarrow S_k(1) \to S_k(2). \qquad (6)$$

Элементы $Q_k \Big| k = \overline{1, card(I_S)}$ могут принимать только два значения: 0 и 1. Первое значение означает, что операция не может выполняться вследствие применения того или иного ограничения. Второе – ограничений на операцию не наложено. Очевидно, что $Q_k = \prod_{c=1}^{card(Q)} q_c^{(k)}$, значение каждого вида ограничений *k*-ой операции также может принимать значения лишь 0 или 1 по аналогичному положению. Формализация рассматриваемых типов ограничений приведена, например, в работе [7].

Интерполяция выражения (4) на весь технологический процесс в целом может быть осуществлена следующим образом:

$$\left.\begin{cases} \exists \Im_k : P_{out}^{(l,k)} \times P_{in}^{(l,k)} \to N_{\Im}, N_{\Im} > 0 \\ Q_k \cdot m^{(k)}(p_i) \geq \left(f^{(k)}(< p_i, t_j >) + b^{(k)}(< p_i, t_j >) \right) \Big| p_i \in P_{in}^{(k)}, \Rightarrow S_k(1) \to S_k(2) \\ Q_k\left(m^{(k)}(p_i)\right) = \prod_{c=1}^{card(Q)} q_c^{(k)} = 1 \end{cases}\right\} \Rightarrow$$

$$\Rightarrow \Re(1) \to \Re(2)$$

Интерпретируется данное выражение следующим образом. Технологический процесс переходит из состояния $\Re(1)$ в состояние $\Re(2)$ вследствие смены состояния операции S_k при условиях:

1) при условии существования функции инцидентности между некоторой операцией S_l и операцией S_k, $l, k \in I_s, l \neq k$, данное условие определяет «причину» смены состояния самой операции S_k;

2) при условии разрешения смены состояния операции S_k;

3) при условии отсутствия ограничений на смену состояний операции S_k.

Таким образом, рассмотрены все элементы выражения (3). Множество Δ_D содержит только те кортежи из множества Δ, которые удовлетворяют множеству отношений Q и имеют в первым элементом $\mathrm{M}_{нач\ out}$, а последним – $\mathrm{M}_{кон\ out}$. Вполне возможно, что мощность множества кортежей Δ_D будет больше 1. В этой связи обоснована постановка вопроса выбора наилучшего варианта выполнения процесса в смысле некоторого множества W заданных показателей качества процесса. Для определения соответствия «результат – показатель» при реализации допустимых вариантов выполнения процесса вводится оператор Ψ [8]:

$$\Psi : \Delta_D \times \mathrm{M}_{нач\ out} \times \mathrm{M}_{кон\ out} \times Q \to W .$$

С учетом всех введенных в рассмотрение множеств и отношений можно сформулировать обобщенную модель Π проблемной ситуации:

$$\Pi : \left\{ \Re, \mathrm{M}_{нач\ out}, \mathrm{M}_{кон\ out}, \Delta_D, W \right\},$$

позволяющую вплотную подойти к постановке задачи выбора оптимального варианта Δ^* выполнения технологического процесса $\Re$ на множестве Δ_D допустимых альтернатив:

$$\Pi \to K : \Delta_D \xrightarrow{W} \Delta^* \tag{7}$$

где K – критерий выбора оптимального варианта выполнения процесса.

На практике множество показателей эффективности содержит более одного элемента. Неодномерность множества показателей эффективности, определяющих выбор оптимального варианта выполнения процесса, порождает задачу многокритериального выбора. Решение данной задачи составляет отдельное направление исследований.

Таким образом, математическая структура выбора может быть представлена в виде:

$$\Delta^* = K\left(\Re, \Delta, Q, R\right), \tag{8}$$

где K – некоторая функция, значением которой является оптимальный кортеж Δ^* разметок входных позиций обобщенных схем операций технологического процесса, а аргументами которой являются следующие множества: $\Re$ – 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 требования и другие требования, $R=\left\{r_i, i=\overline{1, card\left(R\right)}\right\}$ – множество отношений предпочтения (показателей эффективности), определяющих выбор оптимального кортежа Δ^* разметок входных позиций обобщенных схем операций технологического процесса.

3. Обоснование метода динамического программирования

После формирования математической структуры выбора (8) 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 к настоящему времени разработан достаточно мощный методологический аппарат анализа и синтеза, который имеет множество примеров практической реализации в авиационно-космической отрасли [9, 10]. Из всего многообразия методов решения задач такого типа наиболее часто используется принцип оптимальности, положенный в основу метода так называемого динамического программирования [11]. Условиями применения такого метода является следующее.

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 разрабатывается активно в последние годы. 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 $card\left(\mathrm{M}_{out}\right)$ возможных состояний модели процесса и, кроме того, закрепление левого $\mathrm{M}_{нач\ out}$ и правого $\mathrm{M}_{кон\ out}$ 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 (8) результирующим показателем оптимальности является некоторая функция от элементов $r_i, i=\overline{1, card\left(R\right)}$ множества R. Известно [11], что принцип оптимальности является следствием аддитивности показателя оптимальности и не имеет места в случае неаддитивного критерия.

Необходимо подробно рассмотреть интерпретацию метода динамического программирования для случая представления управляемого процесса моделью (2) и структуры выбора множеством (8).

4. Многошаговый процесс управления

Рассмотрим управляемый процесс $\Re$, состояние M_{out} которого характеризуется разметками $M_{k\ out}\Big|k=\overline{1,card\left(I_S\right)}$ выходных позиций P_{out} обобщенных схем $S_k\Big|k=\overline{1,card\left(I_S\right)}$ операций процесса $\Re$. Для определенности примем, что смена состояния процесса происходит с получением отсчетов событий OC (см. рис. 1). Предполагаем, что на каждом шаге на процесс оказывается управляющее воздействие, заключающееся в изменении разметки $M_{in}\left(OC\right)$ входных позиций P_{in} обобщенных схем. Таким образом, в каждый момент OC состояние процесса характеризуется разметкой $M_{out}\left(OC\right)$, а управляющее воздействие – разметкой $M_{in}\left(OC\right)$. Так как нами используются только факты получения OC, то можем считать, что переменная OC 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

$$q\left(OC\right)\in Q,\ OC=0,1,\ldots \tag{9}$$

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

$$M_{out}\left(OC+1\right)=f\left(M_{out}\left(OC\right),M_{in}\left(OC\right)\right). \tag{10}$$

Здесь $f\left(M_{out}\left(OC\right),M_{in}\left(OC\right)\right)$ – 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

$$M_{out}(0) = M_{нач\ out}. \tag{11}$$

Таким образом, многошаговый процесс управления описывается соотношениями (9) – (11). Процедура расчета конкретного процесса сводится к следующему. Пусть с получением некоторого отсчета OC состояние $M_{out}(OC)$ процесса известно. Тогда для определения состояния $M_{out}(OC+1)$ необходимо выполнить две операции: 1) выбрать допустимое управление $M_{in}(OC)$, удовлетворяющее условию (9); 2) определить состояние $M_{out}(OC+1)$ перед получением следующего отсчета OC. Так как начальное состояние процесса известно, то описанную процедуру можно выполнить последовательно для всех $OC = 0, 1, \ldots$

5. Задача оптимального управления

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

$$J = \sum_{OC=0}^{N-1} R\Big(M_{out}(OC), M_{in}(OC)\Big) + F\Big(M_{out}(N-1)\Big). \tag{12}$$

Выражение (12) – это способ представления элемента J структуры выбора (8). Здесь $R\Big(M_{out}(OC), M_{in}(OC)\Big)$ 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 $M_{out}(N-1) = M_{кон\ out}$.

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 (9), и соответствующей траектории, т. е. последовательности (кортежу)

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

которые в совокупности доставляют минимальное (максимальное) значение критерию (12) для процесса (10) и (11). Можно записать

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

6. Элементарный подход

Решить рассматриваемую задачу возможно элементарным подходом или простым перебором. При помощи соотношения (10) состояние процесса после поступления очередного OC выражается через его состояние до поступления и управление, сформированное поступившим OC. Применяя это соотношение многократно, можно выразить состояния процесса после каждого OC только через начальное состояние $M_{нач\ out}$ и управления после предшествующих OC. В результате получим из (12)

$$J = R\left(M_{нач\ out}, M_{in}(0)\right) + R\left(M_{нач\ out}, M_{in}(0), M_{in}(1)\right) + \ldots = \\ = \Phi\left(M_{нач\ out}, M_{in}(0), M_{in}(1), \ldots, M_{in}(N-1)\right).$$

Здесь Φ – некоторая громоздкая, но, вообще говоря, известная функция своих аргументов. Таким образом, поставленная задача оптимального управления свелась к задаче минимизации (максимизации) функции Φ от переменных $M_{нач\ out}$, $M_{in}(0)$, $M_{in}(1)$, …, $M_{in}(N-1)$, т. е. от $10Nk$ переменных, где 10 – количество входных и выходных позиций обобщенной схемы одной операции, k – количество операций в процессе $\Re$. При больших N и k эта задача о минимизации (максимизации) функции большого числа переменных представляет трудности даже при использовании мощных компьютеров. Дополнительное осложнение вызвано тем, что переменные $M_{in}(OC)$ должны удовлетворять ограничениям (9).

Принципиально иной подход к поставленной проблеме дает метод динамического программирования.

7. Принцип оптимальности

Сформулированный Р. Беллманом принцип оптимальности гласит [11]: отрезок оптимального процесса от любой его точки до конца сам является оптимальным процессом с началом в данной точке.

Рассмотрим тривиальное доказательство принципа оптимальности от противного в приложении к рассматриваемой модели технологического процесса. Пусть $M_{out}(OC^*)$ – некоторая точка оптимальной траектории после получения OC^*,

$0 < OC^* < N-1$. Рассуждая от противного, предположим, что отрезок этого процесса от момента получения OC^* до окончания процесса не является оптимальной траекторией для процесса (10) с ограничениями (9), в смысле критерия качества

$$J^* = \sum_{OC=OC^*}^{N-1} R\left(M_{out}(OC), M_{in}(OC)\right) + F\left(M_{out}(N)\right) \tag{13}$$

при начальном условии $M^*_{нач\ out} = M_{out}\left(OC^*\right)$. Значит, существуют допустимое управление $M_{in}\left(OC^*\right), M_{in}\left(OC^*+1\right),..., M_{in}(N-1)$ и соответствующая ему траектория $M_{out}\left(OC^*\right) = M^*_{нач\ out}, M^*_{out}\left(OC^*+1\right), ..., M_{out}(N-1) = M_{кон\ out}$, для которых критерий (13) принимает меньшее (большее) значение, чем на исходном оптимальном процессе. На рис. 1 исходная оптимальная траектория $M_{out}(OC)$ показана линией большей толщины, а траектория $M^*_{out}(OC)$ – меньшей.

Наряду с исходным оптимальным процессом $M_{out}(OC)$, $OC = 0,...,\mathrm{N}-1$, рассмотрим процесс, состоящий из двух участков: части исходного процесса $M_{out}(OC)$, $OC = 0,...,OC^*$, и подозреваемого на «улучшение» участка $M^*_{out}(OC)$, $OC = OC^*,...,\mathrm{N}-1$. Для этого составного процесса критерий J из (12) будет иметь меньшее (большее) значение, чем для исходного процесса, т.к. сумма первых t слагаемых в (12) для составного процесса останется той же, что и для исходного процесса, а сумма остальных слагаемых, равная J^* из (13), уменьшится по сравнению с исходным процессом. Данное утверждение означает, что исходный процесс не является оптимальным, а это противоречит сделанному предположению.

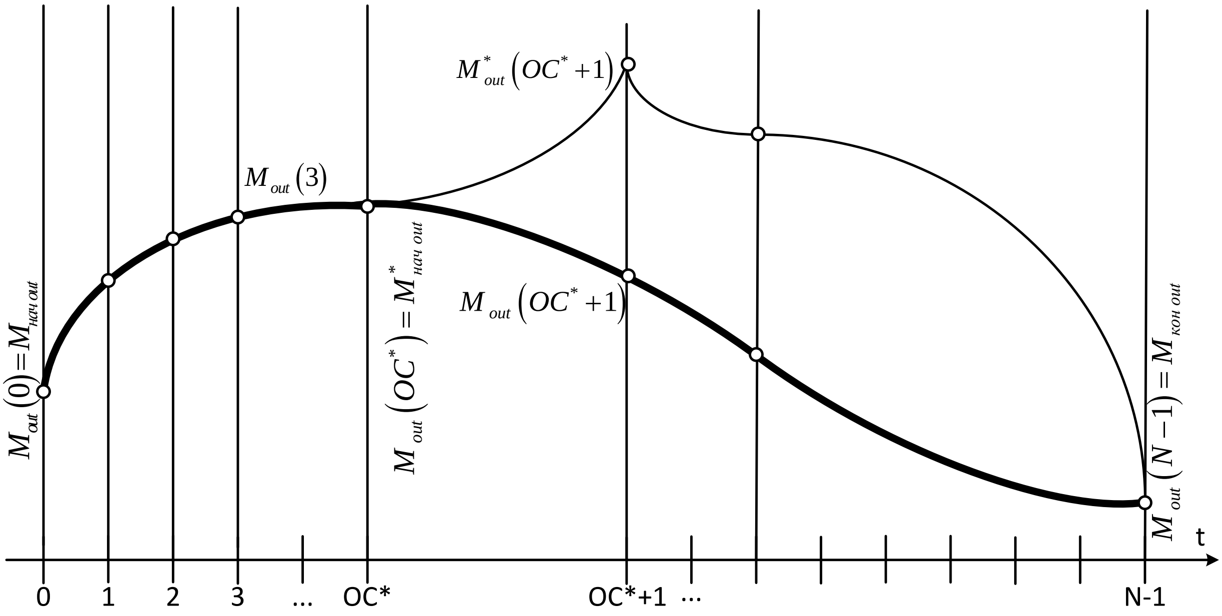

Рисунок 1 – Траектория управляемого процесса

Таким образом, принцип оптимальности доказан.

8. Метод динамического программирования

Обозначим: $J^*_{\min}\left(M_{out}\left(OC^*\right),\mathrm{OC}^*\right)$ – минимальное значение критерия качества J^* из (13) для оптимального процесса, начинающегося при получении OC^* в состоянии $M_{out}\left(OC^*\right)$. Здесь и далее в качестве экстремального значения будет рассматриваться минимальное значение функции (13). Этот процесс можно представить состоящим из двух участков: первого шага, на котором выбирается управление $M_{in}\left(OC^*\right)$, и остальной части (от момента $\left(OC^*+1\right)$ до конца процесса). Вклад в показатель качества первого участка равен $R\left(M_{out}\left(OC^*\right),M_{in}\left(OC^*\right)\right)$, а вклад второго участка можно, согласно принципу оптимальности, выразить через введенную выше функцию $J^*_{\min}$ в виде $J^*_{\min}\left(M_{out}\left(OC^*+1\right),\text{OC}^*+1\right)$. Учитывая, что управление на первом участке должно выбираться из условия минимизации критерия J^* при ограничениях (9), получим равенство

$$J^*_{\min}\left(M_{out}\left(OC^*\right),\text{OC}^*\right)=$$
$$=\min_{M_{in}\left(OC^*\right)\in\text{M}_{in}}\left[R\left(M_{out}\left(OC^*\right),M_{in}\left(OC^*\right)\right)+J^*_{\min}\left(M_{out}\left(OC^*+1\right),\text{OC}^*+1\right)\right].$$

Здесь M_{in} – множество всех возможных состояний разметок входных позиций фрагмента сети, допустимых с учетом (9).

Подставляя в полученное соотношение равенство (10), получим основное соотношение метода динамического программирования

$$J_{\min}\left(M_{out}\left(OC\right),\text{OC}\right)=\min_{M_{in}\left(OC\right)\in\text{M}_{in}}\left[\begin{array}{c}R\left(M_{out}\left(OC\right),M_{in}\left(OC\right)\right)+\\+J^*_{\min}\left(f\left(M_{out}\left(OC\right),M_{in}\left(OC\right)\right),\text{OC}+1\right)\end{array}\right] \quad (14)$$

Для оптимального процесса, начинающегося при получении последнего $OC=N-1,$ критерий оптимальности (13) сводится к одному последнему слагаемому. Поэтому имеем

$$J_{\min}\left(M_{out}\left(N-1\right),N-1\right)=F\left(M_{out}\left(N-1\right)\right)\ . \quad (15)$$

Соотношение (14) и условие (15), играющее роль начального условия, дают возможность последовательно определить функции $J_{\min}\left(M_{out}\left(OC\right),OC\right)$ при $OC=N-1,...,0,$ а также рассчитать оптимальное управление и оптимальные траектории. Это достигается при последовательной реализации обратной и прямой процедур динамического программирования.

9. 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 $OC = N-1$ в (14) и воспользуемся условием (15). Получим

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

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

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

Запись $\upsilon_{N-1}\left(M_{out}\left(OC\right)\right)$ означает, что значение $M_{in}\left(OC\right)$ зависит от $M_{out}\left(N-1\right)$ как от параметра. Определив $J_{\min}\left(M_{out}\left(N-1\right), N-1\right)$ и полагая $OC = N-2$, найдем из (14) функцию $J_{\min}\left(M_{out}\left(N-2\right), N-2\right)$ и соответствующее значение аргумента $M_{in}\left(OC\right) = \upsilon_{N-2}\left(M_{out}\left(OC\right)\right)$.

Продолжая этот процесс в сторону уменьшения OC, получим из (14) последовательно функции $J_{\min}\left(M_{out}(OC), OC\right)$ и

$$\upsilon_{OC}\left(M_{out}(OC)\right) = \underset{M_{in}(OC)\in \mathrm{M}_{in}}{\arg\min}\left[R\left(M_{out}(OC), M_{in}(OC)\right) + J_{\min}\left(M_{out}(OC), OC\right) + 1\right] \quad (16)$$

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

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

10. Прямая процедура

Воспользуемся результатами обратной процедуры для решения исходной задачи, т.е. для построения оптимального управления и оптимальной траектории при заданном начальном условии (11).

Полагая $OC = 0$ и $M_{out}(0) = M_{нач\ out}$ в (16), найдем управление при получении первого OC: $M_{in}(0) = \upsilon_0\left(M_{нач\ out}\right)$. Далее из соотношения (10) определим состояние

$M_{out}(1) = f\left(M_{нач\ out}, M_{in}(0)\right)$. Продолжая этот процесс, найдем $M_{in}(1) = \upsilon_1\left(M_{out}(1)\right)$, $M_{out}(2)$, $M_{in}(2)$ и т. д. Вообще имеем

$$\begin{gathered} M_{in}(OC) = \upsilon_{OC}\left(M_{out}(OC)\right),\ M_{out}(OC+1) = f\left(M_{out}(OC), M_{in}(OC)\right), \\ M_{out}(0) = M_{нач\ out},\ OC = 0, 1, \dots, N-1. \end{gathered} \tag{17}$$

Соотношения (17) определяют прямую процедуру и позволяют полностью рассчитать оптимальное управление и оптимальную траекторию. Минимальное (максимальное) значение критерия оптимальности, отвечающее этой траектории, $J = J_{\min}\left(M_{нач\ out}, 0\right)$.

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.

11. Проклятие размерности

При реализации этого метода получен значительно более общий и универсальный результат. В ходе обратной процедуры построено оптимальное управление по обратной связи (16), т. е. управление как функция $\upsilon_{OC}\left(M_{out}(OC)\right)$ текущего состояния процесса. Теперь нетрудно дать решение задачи оптимального управления при любом начальном условии вида $M^*_{нач\ out} = M_{out}\left(OC^*\right)$: для этого нужно просто реализовать прямую процедуру для этого начального условия. Реализация прямой процедуры не предоставляет серьезных трудностей и сводится, согласно (17), к вычислению известных функций при конкретных значениях аргументов.

Наибольшую сложность представляет обратная процедура, включающая минимизацию функции J по аргументу $M_{in}(OC)$. Размерность аргумента $M_{in}(OC)$ определяется величиной $card(I_S)$ – количеством операций в процессе, а также количеством входных позиций в одной операции $S_k \big| k=\overline{1,card(I_S)}$. Определенно можно сказать только о количестве входных позиций: обобщенная схема операции содержит пять входных позиций. Поэтому размерность аргумента $M_{in}(OC)$ равна произведению 5 и $card(I_S)$. Итак, здесь нужно, согласно (14), выполнить N процедур минимизации функции от $(5\cdot card(I_S))$ переменных. Это значительно более простая задача, чем минимизация одной функции J по $(N\cdot 5\cdot card(I_S))$ переменным, что требуется при элементарном подходе.

Однако указанная простота не сводит осуществление обратной процедуры к тривиальной задаче. Дело в том, что обратная процедура предусматривает построение функций $J_{\min}(M_{out}(OC),OC)$ и $\upsilon_{OC}(M_{out}(OC))$, зависящих от аргумента $M_{out}(OC)$. Размерность аргумента $M_{out}(OC)$ точно такая же, как и размерность аргумента $M_{in}(OC)$ при осуществлении обратной процедуры. По аргументу $M_{out}(OC)$ не нужно выполнять операций минимизации, но даже простое табулирование и хранение функций $(5\cdot card(I_S))$ переменных представляют большие трудности. Например, если количество OC, ожидаемых к поступлению, ограничено 10 значениями, то, учитывая, что из выходных позиций обобщенной схемы две являются счетными, а три – бинарными, обратная процедура, в которой участвуют функции $J_{\min}\left(M_{out}\left(OC\right),OC\right)$ и $\upsilon_{OC}\left(M_{out}\left(OC\right)\right)$, потребует $5\cdot card\left(I_S\right)\cdot N\cdot\left(10\cdot10\cdot10\cdot2\cdot2\right)^{card\left(I_S\right)}$ ячеек памяти. В указанном количестве необходимых ячеек памяти первый и второй множители определяются размерностью аргумента $M_{in}\left(OC\right)$, третий – длительностью процесса в единицах OC, четвертый – количеством вариантов разметки выходных позиций всех обобщенных схем.

Отсюда понятно, что вычислительная реализация метода динамического программирования сталкивается с большими трудностями. Эти трудности Р. Беллман назвал проклятием размерности. Очевидно, что данное затруднение сопровождает применение метода динамического программирования и для оптимизации технологического процесса, модель которого построена на обобщенной схеме операции. Для преодоления этих трудностей могут быть предложены подходы [12], 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

Развитие информационных технологий на сегодняшний момент позволяет реализовать концепцию Единого информационного пространства [13], в том числе и в такой технически и организационно сложной области деятельности, как космическая деятельность. Это соответствует перспективным направлениям развития ракетной отрасли. Одной из основных проблем при этом является автоматизация оптимального управления технологическими процессами.

В статье рассмотрено общее содержание проблемы автоматизированного управления технологическими процессами. При этом с позиций системного подхода сама проблема структурирована в виде 5 основополагающих элементов, требующих обязательного рассмотрения при автоматизации: цели, ограничения, альтернативы и критерии принимаемых решений при автоматизированном управлении, а также модели управляемого процесса.

В качестве одного из вариантов модели управляемого процесса предлагается модель на основе модифицированной сети Петри, созданная на основе структурно-логического подхода. В соответствии с этим подходом структура процесса представляется в виде совокупности неделимых операций с наложенными связями. Каждая операция представляет собой неизменную (универсальную) сеть, построенную на основе G-сетей. Связи между операциями являются отображением логики процесса и всех ограничений выполнения технологического процесса.

Применительно к предложенной модели управляемого процесса сформулирована математическая структура выбора оптимального варианта выполнения процесса. Введены необходимые множества и отношения с адаптацией к применению структурно-логического подхода к моделированию процесса. Показано, что задача выполнения технологического процесса может быть поставлена как оптимизационная задача многокритериального выбора варианта его выполнения на множестве допустимых альтернатив. Полученная математическая структура выбора имеет предельно общий характер и при практическом применении требует конкретизации введенных множеств и отношений между ними.

К настоящему времени разработан достаточно мощный методологический аппарат анализа и синтеза для решения оптимизационной задачи управления в однокритериальной постановке. В статье рассмотрены особенности применения метода динамической однокритериальной оптимизации при использовании структурно-логического подхода к моделированию технологического процесса. В качестве основы используется метод динамического программирования.

В дальнейшем представляется важным рассмотрение особенностей применения методов многокритериальной оптимизации при аналогичных условиях.

Библиографический список

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 // Труды МАИ, 2015, № 84: http:

3. Котов В.Е. Сети Петри. - Л.: Наука, 1984. - 160 с.

4. Охтилев М.Ю. Основы теории автоматизированного анализа измерительной информации в реальном времени. Синтез системы анализа: Монография. – СПб.: ВКА имени А.Ф. Можайского, 1999. - 162 с.

5. URL: http://cpntools.org/download.

6. Kristensen L.M., Christensen S., Jensen, K. The Practitioner's Guide to Coloured Petri Nets. Int. J. Softw. Tools Technol. Transf. 2(2), 98–132 (1998).

7. Шмелев В.В., Мануйлов Ю.С., Рахимов Р.Р., Богданов А.В. Формализация технологического процесса на основе сетевой модели // Научное обозрение. 2015. №19. С. 156 – 161.

8. Резников Б.А. Системный анализ и методы системотехники. Часть 1. Методология системных исследований, моделирование сложных систем. - М.: МО СССР, 1990. - 522 с.

9. Закиров Р.Г. Оптимизация алгоритмов диагностики состояния бортового радиоэлектронного оборудования // Труды МАИ, 2014, № 78: http: http://www.mai.ru/science/trudy/published.php?ID=53665.

10. Родионова Д.А. Синтез оптимальных детерминированных систем с полной обратной связью методом итерационного динамического программирования // Труды МАИ, 2015, № 84: http: http://www.mai.ru/science/trudy/published.php?ID=63137

11. Черноусько Ф.Л. Динамическое программирование // Соросовский образовательный журнал. 1998. №2. С. 139 – 144.

12. Управление космическими аппаратами и средствами наземного комплекса управления / Под ред. Ю.С. Мануйлова – СПб.: ВКА имени А.Ф. Можайского, 2010. - 609 с.

13. Сырин С.А., Терещенко Т.С., Шемяков А.О. Анализ прогнозов научно-технического развития России, США, Китая и Европейского Союза как лидеров ракетно-космической промышленности // Труды МАИ, 2015, № 82: http://www.mai.ru/science/trudy/published.php?ID=58745