

Таким образом, в данной статье был рассмотрен метод DEA, предложена и построена граница отставания, а также представлено разбиение множества объектов на два класса, что облегчает ЛПР классификацию объектов, так как предполагает существование всего двух классов: плохого и хорошего. Это удобно, когда выборка объектов достаточно велика. Если ЛПР требуется точное количественное значение эффективности, то он получает это решение из прямого решения задачи по методу DEA.

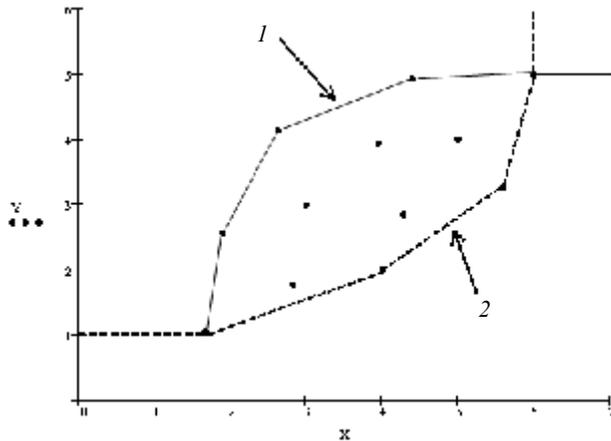


Рис. 3. Графики границы эффективности (I) и границы отставания (II)

Библиографический список

1. Моргунов, Е. П. Многомерная классификация на основе аналитического метода оценки эффективности сложных систем / Е. П. Моргунов ; НИИ систем упр., волновых процессов и технологий. Красноярск, 2003.
2. Coelli, T. An introduction to efficiency and productivity analysis / T. Coelli, D. S. Prasada Rao, G. E. Battese. Boston : Kluwer Academic Publishers, 1998.

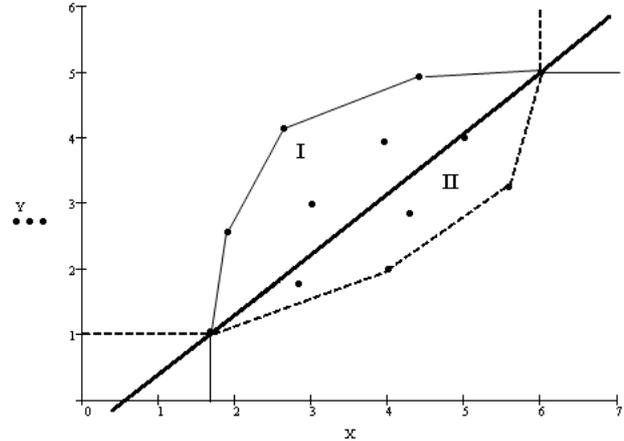


Рис. 4. Классы объектов, полученные разбиением совокупности объектов с помощью разделяющей прямой

Т. А. Рукавицына

MODEL DEVELOPMENT OF THE DEA METHOD

DEA (Data Envelopment Analysis) method and its modification are considered. An inefficient frontier is suggested to be introduced. The intersection of the efficient frontier and inefficient frontier are investigated. A hyper plane going through the intersection of both frontiers is constructed to divide the sample into two classes.

Keywords: DEA, efficiency, efficient frontier.

© Рукавицына Т. А., 2009

УДК 681.34

А. А. Ступина, Е. С. Разгулина

GERT-АНАЛИЗ ФОРМИРОВАНИЯ ТЕХНОЛОГИЧЕСКИХ ЦИКЛОВ УПРАВЛЕНИЯ КОСМИЧЕСКИМИ АППАРАТАМИ

Предложено использовать аппарат сетевого анализа, включая GERT-анализ, для решения задач оптимизации формирования технологических циклов управления (ТЦУ).

Ключевые слова: GERT-сети, GERT-анализ, ациклическая сетевая модель технологических циклов управления, оптимизационная модель, вероятностные характеристики ТЦУ, время реализации ТЦУ.

Решение задачи формирования технологических циклов управления (ТЦУ) связано в первую очередь с необходимостью обеспечить взаимодействие с объектом (космическим аппаратом) в соответствии с временными циклограммами в режиме реального времени. Более простая форма режима реального времени предполагает лимитирование

времени ответа центра управления полетами (ЦУП) на запрос от объекта. Ограничения на время реакции связываются в этом случае с выполнением периодических действий в рамках технологического цикла управления.

Естественной математической интерпретацией распределенных, асинхронных, мультипроцессорных и мульт-

типрограммных систем являются сетевые модели, которые позволяют отражать распределенность структуры, сетевой характер взаимосвязей между процессами и ресурсами, аппаратными и программными компонентами средств управления. В связи с этим для решения задач оптимизации формирования ТЦУ целесообразно привлекать аппарат сетевого анализа, включая GERT-анализ (GERT – аббревиатура от *graphical evaluation and review technique*). В последнее время GERT-сети получают все большее распространение для моделирования и оптимизации технических систем, к числу которых можно отнести ТЦУ космических аппаратов.

Общность методов построения моделей ТЦУ для анализа и проектирования, выражающаяся в использовании сетевых моделей, позволяет создать единые инструментальные средства оптимизации формирования ТЦУ.

Сетевое представление ТЦУ открывает широкие возможности для использования в моделях анализа реализуемости циклограмм математического аппарата, учитывающего вероятностную интерпретацию. Одна из таких возможностей связана с введением неопределенности в продолжительность реализации задач ТЦУ (аналогично использованию PERT-методики в сетевом анализе). Однако такая модель представляет собой типичную детерминированную сеть, для полного выполнения которой необходимо выполнение всех дуг, т. е. безоговорочное выполнение соответствующих операций ТЦУ. Из этого условия следует, что в такую модель не могут быть включены операции с обратной связью, поскольку они представлены петлями, существование которых в свою очередь означает, что конечный узел операции должен быть выполнен раньше ее начального узла.

Одним из путей оптимизации формирования ТЦУ является использование GERT-анализа, который позволяет учесть риск изменения состава задач при наступлении определенных событий или по результатам выполнения предшествующих задач. Возможность применения GERT-анализа также связана с использованием сетевых моделей со стохастической структурой, так как нередко именно они оказываются наиболее гибкими и полезными.

Согласно [1], узлы стохастической сети могут быть интерпретированы как состояния процесса, а дуги – как переходы из одного состояния в другое. Такие переходы можно рассматривать как реализацию обобщенных задач ТЦУ, характеризующих плотностью распределения, или функцией массы, и вероятностью выполнения. В результате получается стохастическое графовое представление ТЦУ, где узлы являются входом и выходом для очередной задачи, а дуги характеризуют время выполнения реальной технологической операции.

Каждый внутренний узел стохастической сети выполняет две функции, одна из которых касается входа в узел, а другая – выхода. Обычно эти функции называют входной и выходной.

В [1] определен следующий тип входной функции: узел выполняется, если выполнена дуга, входящая в него, при условии, что в заданный момент времени может выполняться только одна дуга.

Для выхода там же определены два типа выходной функции: детерминированный выход и вероятностный

выход. Для детерминированной выходной функции характерна ситуация, когда все дуги, выходящие из узла, выполняются, если этот узел выполнен. У вероятностной выходной функции ровно одна дуга, выходящая из узла, выполняется, если узел выполнен. Выбор такой дуги может быть описан с помощью функции распределения вероятностей.

При анализе реализуемости ТЦУ стохастическая сеть определяется как сеть, которая работает только при выполнении некоторого подмножества дуг, при этом время выполнения каждой дуги (задачи ТЦУ) выбирается в соответствии с вероятностным распределением. В такой стохастической сети для выполнения узла нет необходимости в выполнении всех дуг, входящих в него. Поэтому в такой модели допускается существование циклов и петель.

Для более наглядного представления детерминированного случая можно обратиться к простому ациклическому детерминированному ТЦУ, который имеет GERT-подобную узловую логику. Такая модель соответствует сети для проектирования (решения) ТЦУ. Это указывает на то, что осуществляется выбор, т. е. принимается решение о том, какие задачи ТЦУ должны быть выполнены для минимизации некоторой целевой функции. Для учета вероятностных характеристик реализации ТЦУ вводится понятие случайных акций и рассматривается возможность многократной последовательной реализации ТЦУ до момента успешного завершения.

Пусть N – ациклическая сетевая модель ТЦУ с источниками и стоками (действия, соответствующие задачам ТЦУ, представляются дугами), где множество узлов обозначается V , а множество дуг – E . Предположим, что N имеет только один исток, который обозначается через r и соответствует началу рассматриваемого ТЦУ. Также предположим, что один из стоков N представляет собой успешное завершение ТЦУ. Обозначение этого стока – s . Оставшиеся стоки, если они есть, могут представлять собой различные виды неудачного завершения или прерывания процесса управления.

Определение 1. Ациклическую сетевую модель ТЦУ $N(V, E)$ только с одним истоком и со стоками назовем сетью для проектирования (решения) ТЦУ, если каждый узел i из N определен через входную характеристику $CH_i \in \{0, 1, \dots, |P(i)|\}$ и выходную характеристику $CH_i^+ \in \{0, 1, \dots, |S(i)|\}$, где множество узлов обозначается V , а множество дуг – E ; $|P(i)|$, $|S(i)|$ – мощность множества предшественников и последователей узлов i соответственно. Эти характеристики, формирующие узловую логику, имеют следующие значения:

- а) узел активируется сразу же, как только входные действия CH_i завершаются;
- б) как только узел i активирован, то не более CH_i^+ выходных действий начинает выполняться. Если узел i не активируется, то ни одно выходное действие не выполняется. Иногда уместно заменить сочетание «не более» на «точно».

Два условия из определения 1 подразумевают, что каждое действие выполняется сразу как только это становится возможным.

Для источника r полагаем $CH_r = 0$, т. е. он всегда активирован. Кроме того, $CH_i^+ = 0$ для $i \in S$, где S – множество стоков N .

Заметим, что если $CH_i^+ = 1$, то узел i имеет OR-вход, а если $CH_i^- = |P(i)|$, то тогда i имеет AND-вход. И если в определении 1б «не более» заменяется на «точно», то $CH_i^+ = 1$ соответствует вероятностному выходу, а $CH_i^- = |S(i)|$ – детерминированному выходу. Если же заданная сеть N для формирования ТЦУ имеет множество источников R ($|R| > 1$) и множество $R' \cap R, R' \neq \emptyset$ активизируется в начале выполнения ТЦУ, то можно формально перевести N в соответствующую одноисточковую сеть.

Дуговые переменные обозначаются через w_{ij} , при этом полагается, что $w_{ij} = 1$, если (i, j) выполняется ($(i, j) \in E$), и $w_{ij} = 0$ в противном случае. Узловые переменные $u_i = 1$ ($i \in V$), если i активизируется, иначе $u_i = 0$, где $u_r = 1$, т. е. источник всегда активизируется. Тогда условия узловой логики, введенные в определении 1, могут быть представлены в следующем виде:

$$\sum_{k \in P(i)} w_{ki} \geq CH_i^+ u_i; (i \in V \setminus \{r\}), \quad (1)$$

$$\sum_{k \in P(i)} w_{ki} < CH_i^- + M_i u_i; (i \in V \setminus \{r\}), \quad (2)$$

где $M_i > |P(i) - CH_i^-|$,

$$\sum_{j \in S(i)} w_{ij} \leq CH_i^+ u_i; (i \in V \setminus S). \quad (3)$$

Так как решающая модель ациклична, то каждая задача соответствующего ТЦУ либо выполняется только один раз, либо не выполняется вообще. Таким образом, каждая реализация ТЦУ может быть соотнесена с множеством выполняемых задач ТЦУ или с функцией $w: E \rightarrow \{0, 1\}$; $((i, j) \in E)$, значения которой задаются как $w((i, j)) = w_{ij} = 1$, если (i, j) выполняется, и 0, если иначе.

С другой стороны, если некоторая w -я реализация ТЦУ задана, то как узловые, так и дуговые переменные для этого случая специфицированы и можно говорить о допустимой реализации ТЦУ, если w удовлетворяет условиям узловой логики. Тогда $e = \{w: E \rightarrow \{0, 1\} | w_{ij} \text{ удовлетворяет условиям (1)–(3); } (i, j) \in E\}$ и e – множество всех допустимых реализаций ТЦУ.

Если в решающей сети для формирования ТЦУ обозначить вес дуги $(i, j) \in E$ как длительность $d_{ij} \in R_+$ соответствующей задачи ТЦУ, то d_w – длительность w -й реализации ТЦУ, т. е. время, требуемое для исполнения всех задач (i, j) при $w_{ij} = 1$. Необходимо минимизировать d_w при условиях: w активизирует s ; ($w \in \varepsilon$). Через $\varepsilon^* = \{w \in \varepsilon | w \text{ активизирует } s\}$ обозначим множество успешных реализаций ТЦУ. Для $\varepsilon^* \neq \emptyset$ $d^* = \min_{w \in \varepsilon^*} d_w$ соответствует минимальному значению целевой функции задачи.

Приняв, что каждая реализация ТЦУ начинается с задерживания истока r в момент 0, считаем для $w \in \varepsilon$, что t_j^w есть время активации узла $j \in V$ для w -й реализации, причем $t_r^w = 0$ и $t_j^w = \infty$, если j не активизируется в течение w -й реализации ТЦУ. Для $j \in V \setminus \{r\}$ имеем $t_j^w = \min\{t \geq 0 | \text{существуют } CH_j^-, \text{ отличные от } i \in P(j) \text{ и такие, что } w_{ij} = 1 \text{ и } t_j^w + d_{ij} \leq t\}$.

Кроме того, справедливо $d_w \geq t_s^w \forall w \in \varepsilon^*$. Тогда $d_w > t_s^w$, если сток s активизируется, пока некоторые действия (i, j) при $w_{ij} = 1$ все еще выполняются. Так как $t_j^e = \min_{w \in \varepsilon^*} t_j^w$ – самое раннее из возможных времен задерживания узла j в течение какой-либо возможной реализации ТЦУ, то

$$t_s^e = \min_{w \in \varepsilon^*} \max_{(i, j) \in E} \{(t_i^w + d_{ij}) w_{ij}\},$$

$$E_w = \{(i, j) \in E | w_{ij} = 1\},$$

где E_w – множество задач, выполняемых в течение w -й реализации ТЦУ.

Учитывая вышесказанное, можно утверждать, что минимальная длительность успешной реализации ТЦУ равна самому раннему моменту задерживания стока s : $d^* = t_s^e$ для $\varepsilon^* \neq \emptyset$. Таким образом, можно найти величину d^* , вычисляя минимально возможные моменты задерживания t_s^e ($j \in V$).

Рассматривая моменты t_i как компоненты вектора временной развертки (ВВР) ТЦУ, которые удовлетворяют условию $(t_j - t_i - d_{ij}) w_{ij} \geq 0, (i, j) \in E; t_i \geq 0, i \in V \setminus \{r\}; t_r = 0$, можно утверждать, что для некоторой w -й реализации ТЦУ ($w \in \varepsilon^*$) моменты t_i^w отвечают этим ограничениям, а самые ранние моменты t_i^e удовлетворяют им для соответствующей минимальной $w \in \varepsilon^*$. Соответствующая оптимизационная задача имеет вид: минимизировать $\max_{(i, j) \in E} \{(t_i^w + d_{ij}) w_{ij}\}$ при выполнении условий (1)–(3) и ограничений ВВР ($u_s = 0$).

По сравнению с минимизацией затрат на выполнение ТЦУ здесь имеются не только добавочные ограничения, но и более сложная целевая функция. Кроме того, дополнительно к узловым и дуговым переменным существуют моменты t_i ; ($i \in V$), которые также являются переменными данной оптимизационной задачи.

С учетом алгоритмически заданных ограничений для вектора временной развертки разработаны процедуры для этапа анализа реализуемости и коррекции ТЦУ, который соответствует общей задаче различимости и включает исследование совместных свойств процессов управления и аппаратно-программных средств с заданным вектором реализации. Согласно этим процедурам, необходимо многократно повторять последовательные этапы оптимизации целевого функционала, а при выборе оптимальной по времени реализации ТЦУ следует учитывать состав программно-аппаратных средств системы. Общая аналитико-оптимизационная процедура, кроме этапа анализа и коррекции, включает такие шаги, как выбор для заданного ТЦУ (или совокупности ТЦУ) оптимального состава мультиверсий для программных модулей с применением приближенных методов, метод критического пути для субоптимальной по времени реализации ТЦУ и точной оптимизации ВВР. Оптимизационные модели формирования мультиверсионного программного обеспечения (ПО) реализации ТЦУ включают как надежные постановки задач для одно- и многофункциональных программных систем, так и многокритериальные постановки.

Отметим, что в рамках разработанной интерактивной системы формирования ТЦУ между всеми этапами решения задачи организован диалоговый интерфейс, который предоставляет пользователю (специалисту по технологии управления) достаточную информацию для принятия решения, причем система способна к расширению. Кроме того, обеспечен стохастический анализ ТЦУ для нахождения математического ожидания и стандартного отклонения директивного времени на реализацию ТЦУ в условиях неопределенности. Для сложного процесса, которым является технологический цикл, директивное время рассматривается как случайная величина с конечным

математическим ожиданием и дисперсией, описанная подходящей функцией распределения. Для получения дисперсионных оценок необходимы некоторые предположения, касающиеся стохастических характеристик каждого элемента структуры ТЦУ (задач обработки информации и управления) в стандартных и, если необходимо, нештатных ситуациях. Такое описание задач ТЦУ по сравнению со случаем, когда заданы только временные характеристики, является более сложным, но, очевидно, более точным.

Процедура, используемая для анализа вероятностных характеристик ТЦУ, базируется на стохастических сетях типа GERT и совмещает теорию потоков в графах, функции генерации момента и PERT-анализ для получения результата. Стохастическое представление ТЦУ в виде GERT-сети позволяет получить достаточное количество полезной информации о временных характеристиках реализации ТЦУ. Используя неравенство Чебышева, можно показать пределы, в которых будет изменяться фактическое время реализации ТЦУ, а также получить более сильные утверждения. Кроме того, если реальный показатель не соответствует этим оценкам, то строится ряд критериев для проверки гипотез, позволяющих определить перспективные характеристики времени реализации ТЦУ.

Рассмотренная GERT-модель является своеобразной альтернативой традиционным методам определения директивных времен реализации ТЦУ, поскольку система моделей позволяет учесть случайные отклонения и неопределенность, возникающие непосредственно во время выполнения каждой отдельной задачи ТЦУ. Следовательно, в полученный результат уже включены все случайные колебания и нет необходимости вносить в него дополнительные поправки, не считая тех, которые соответствуют нештатным ситуациям или авариям.

Анализ моделей формирования ТЦУ показывает, что они имеют, как правило, алгоритмически заданный критерий, а часто – и ограничения. Очевидно, что при такой постановке задачи имеет смысл говорить только об эвристических процедурах случайного поиска.

Формально указанные оптимизационные модели могут быть сведены к задаче псевдобоулевой оптимизации вида

$$F(\mathbf{X}) \rightarrow \text{extr}, \quad (4)$$

$$S = \left\{ X \in B_2^n : \sum_{j=1}^n x_j = m, m < n \right\}, \quad (5)$$

на решение которой ориентированы алгоритмы метода изменяющихся вероятностей (МИВЕР) [2].

Алгоритмы МИВЕР не укладываются в традиционную схему алгоритмов случайного поиска для непрерывной оптимизации, которая в общем случае включает два этапа:

– первый этап – в соответствии с некоторым правилом и вероятностным распределением на множестве всех возможных направлений поиска случайным образом выбирается одно из направлений;

– второй этап – по заданному алгоритму выбирается длина шага перемещения поиска в выбранном направлении.

Для реализации этой схемы в пространстве булевых переменных необходимо определить понятия направления в B_2^n .

Определим инверсию p_i ($1 \leq i \leq n$) как подстановку на множестве B_2^n . И пусть $X, Y \in B_2^n$. Тогда направлением $K(X, Y) \subseteq B_2^n$ назовем множество таких точек $Z \in B_2^n$, которые можно представить в виде

$$Z = \pi_{i_1} \cdot \pi_{i_2} \cdot \dots \cdot \pi_{i_k} X,$$

где $\pi_{i_m} \in W(X, Y)$, $W(X, Y) = \{\pi_{i_1}, \pi_{i_2}, \dots, \pi_{i_k}\}$, такое, что

$$Y = \pi_{i_1} \cdot \pi_{i_2} \cdot \dots \cdot \pi_{i_k} X \quad (k \leq n).$$

В соответствии с этим определением для любой точки $X \in B_2^n$ существует $(2^n - 1)$ направлений, каждое из которых однозначно определяется любой точкой $Y \in B_2^n$ либо множеством $W(X, Y)$. Длину шага по направлению $K(X, Y)$ определяем, используя метрику Хэмминга.

Для введенных таким образом понятий направления и длины шага общая схема построения алгоритмов направленного поиска выглядит следующим образом.

1. Случайным образом выбирается начальная точка поиска X^0 . В соответствии с заданным алгоритмом выбирается точка Y и формируется множество инверсий $W(X^0, Y)$, определяющих выбираемое направление.

2. По заданному правилу определяется длина шага $l \leq \text{card}(W(X^0, Y))$. Формируется множество инверсий $W(X^0, Z) \subseteq W(X^0, Y)$, $\text{card}(W(X^0, Z)) = l$. Осуществляется переход в точку

$$Z = \pi_{i_1} \cdot \pi_{i_2} \cdot \dots \cdot \pi_{i_l} X^0, \pi_{i_j} \in W(X^0, Z).$$

Критерием останова работы алгоритма, как и при случайном поиске на множестве R^n , может быть ограничение по числу итераций, определение точки локального минимума и т. п. Предложенная схема четко подразделяется на два этапа: определение направления поиска и выполнение шага в выбранном направлении.

Приведенные выше процедуры направленного случайного поиска были реализованы программно и прошли апробацию при решении задач формирования ТЦУ. Численные результаты показали, что в ряде случаев алгоритмы направленного поиска позволяют получить лучшие результаты по сравнению с алгоритмами МИВЕР.

Регулярные процедуры в отличие от рассмотренных рандомизированных гарантируют получение точного решения и позволяют априори оценить трудоемкость его получения, тем более что анализ оптимизационных моделей показывает, что в ряде случаев критерии и ограничения задаются явно и, более того, целевые функции являются сепарабельными. В первую очередь это касается задач формирования мультиверсионного ПО.

Сепарабельные псевдобоулевые функции определяются следующим образом:

$$f(X) = f(x_1, \dots, x_n) = \sum_{1 < j < n} g_j(x_j),$$

где $f: B_2^n \rightarrow R^1$.

Оценкой сверху числа необходимых вычислений значений целевой функции, когда поиск начинается в точке $(1, \dots, 1)$ (самой дальней от минимума), будет $n^2 - n + 2$ вычислений, а оценкой в среднем (по выбору начальной точки поиска) будет величина $(n^2 + 4)/2 - 1/2^n$. Данные оценки позволяют сравнить эффективность локального поиска на классе сепарабельных псевдобоулевых функций с эффективностью других подходов, например эволюционных алгоритмов.

Рассмотренный выше алгоритм локального поиска реализует информационную сложность класса слабо немонотонных псевдодобулевых функций. При оптимизации произвольных функций, в том числе и унимодальных, он не дает никаких гарантий за исключением полного перебора.

Библиографический список

1. Филлипс, Д. Методы анализа сетей / Д. Филлипс, А. Гарсия-Диас. М. : Мир, 1984.
2. Kovalev, I. Optimization Problems when Realizing the Spacecrafts Control / I. Kovalev // *Advances in Modeling and Control*. 1998. Vol. 52, № 1, 2. P. 63–70.

A. A. Stupina, E. S. Razgulina

GERT-ANALYSIS OF THE TECHNICAL CONTROL CYCLES FORMATION FOR SPACE CRAFTS

In this paper authors recommend to use network analysis method (including GERT analysis) to optimize formation of technical control cycles (TCC).

Keywords: GERT-networks, GERT-analysis, acyclic network model for TCC, optimization model, TCC relative parameters, time for TCC accomplishment.

© Ступина А. А., Разгулина Е. С., 2009