

УДК 004: 519.854

К.В. КРОТОВ, А.В. СКАТКОВ

Севастопольский национальный технический университет, Украина

ОБОСНОВАНИЕ МНОГОКРИТЕРИАЛЬНОЙ МНОГОУРОВНЕВОЙ МОДЕЛИ ПОСТРОЕНИЯ РАСПИСАНИЙ ГРУППОВОЙ ОБРАБОТКИ ПАРТИЙ ТРЕБОВАНИЙ

Для постановки задач построения расписаний обработки требований в многостадийных конвейерных системах необходимо формирование информационной модели соответствующей обрабатывающей системы, в которой формализуются виды входных данных, виды ограничений на множество решений задачи, выделяемые для обработки требований ресурсы. В работе вводятся обозначения, необходимые для формализации информационной модели конвейерной системы, обосновывается необходимость перехода к иерархической задаче построения расписаний (использование аппарата иерархического программирования). Обосновывается в общем виде многокритериальная модель многоуровневого программирования для формирования эффективного состава групп партий требований и расписаний обработки партий при наличии ограничений на используемые ресурсы вычислительной системы.

Ключевые слова: многоуровневое программирование, партии требований, группы партий, комплексные расписания, многостадийная система, ограничения на используемые ресурсы.

Введение

Повышение производительности выполнения программ при обработке данных достигается путем их (программ) конвейеризации. Конвейеризация программ предполагает разделение их (программ) на фрагменты, каждый из которых закреплен для выполнения за соответствующим сегментом конвейера. При интерпретации программного кода фрагмента на закрепленном для него сегменте конвейера им реализуется обработка соответствующих данных. В связи с реализацией конвейеризованного выполнения программ (фрагментов программ на соответствующих сегментах конвейера) возникает задача составления расписаний пакетной обработки данных этими программами. Пакетом заданий на обработку данных является совокупность программ, выполняющихся в конвейерной системе (n программ), и данных, обрабатываемых ими. Обработка пакетов заданий осуществляется в вычислительной системе при использовании ее ресурсов. Основными ресурсами системы, выделяемыми выполняемым программам, являются время функционирования системы и объем оперативной памяти на каждом из приборов, выделяемый для хранения данных, обрабатываемых программами. Введем в рассмотрение индекс ресурса системы, обозначив его через e , тогда при $e=1$ рассматриваемый ресурс–используемое при выполнении программ процессорное время, при $e=2$ – объем выделяемой для хранения данных оперативной памяти. При этом

количество имеющихся в распоряжении системы ресурсов, используемых ею при выполнении программ, является ограниченным. Сегменту конвейера в рассматриваемой конвейерной системе соответствует понятие «обрабатывающего прибора» в теории расписаний (вычислительное устройство, реализующее выполнение программного кода соответствующей части конвейеризированной программы). Через i обозначим номер множества однотипных данных, характеризующих одинаковые объекты, которые должны быть обработаны в системе, через I – множество всех данных, которые будут обработаны в вычислительной системе (количество множеств данных обозначим через n , $i = \overline{1, n}$). Через n^i обозначим количество элементов в множестве однотипных данных, характеризующих индексом i (множество содержит данные об n^i одинаковых объектах). Тогда в системе должно быть обработано n множеств однотипных данных, каждое из которых содержит n^i элементов. При этом данные, входящие в некоторое i -е множество, обрабатываются соответствующей ему программой. Тогда через i обозначим индекс (номер) программы, выполняемой в составе конвейера и обрабатывающей данные из i -го множества. Если множество данных, характеризующее индексом i , содержит n^i элементов, то программа, реализующая обработку данных i -го множества, должна быть выполнена n^i раз. Через l обозначим индекс (номер) обрабатывающего прибора (сегмента),

входящего в состав конвейера (конвейеризированное выполнение программ предполагает упорядоченность обрабатывающих приборов (сегментов конвейера) в соответствии с их номерами). Количество приборов в конвейере обозначим через L , тогда $l = \overline{1, L}$.

Реализация в многостадийной вычислительной системе выполнения конвейеризированной программы предполагает осуществление вычислений на каждом сегменте конвейера в соответствии с определенной ее (программы) частью. Т.к. l -ый сегмент вычислительной конвейерной системы осуществляет обработку назначенной для него l -ой части i -ой программы, тогда введенному в теории расписаний понятию обрабатываемого требования соответствует выполняемая в вычислительной системе конвейеризированная программа (обработке на l -ом приборе некоторого i -го требования соответствует выполнение l -ой части i -ой конвейеризированной программы). Если в системе выполняется обработка однотипных требований (требований различных i -ых типов, $i = \overline{1, n}$), то из обрабатываемых требований одинаковых типов должны быть сформированы партии (партия – это совокупность требований одного типа, ее характеристиками являются: тип i требований в этой партии и количество требований этого типа в ней). Партия является фиксированной, если в нее входят все требования одного i -го типа. Партия может содержать не все требования i -го типа, в этом случае в системе будет обрабатываться ни одна, а несколько партий требований этого типа. В соответствии с введенными обозначениями в конвейерной системе должны быть обработаны партии однотипных требований при наличии ограничений на ресурсы системы. Управление обработкой пакета заданий предусматривает построение расписаний обработки партий требований в многостадийной конвейерной системе при наличии ограничений на выделяемые системой для этой обработки ресурсы.

Анализ публикаций. Использование современных методов решения задачи формирования расписаний обработки партий требований различных типов позволяет определять порядок обработки фиксированных партий либо определять эффективное количество, состав и порядок обработки партий на ограниченном количестве приборов (на одном приборе) [1]. Постановка задачи обработки партий в общем виде предполагает задание: произвольного количества приборов, ограничений на используемые при обработке требований партий ресурсы системы. Это обуславливает необходимость перехода к формированию эффективного количества и составов партий требований, к формированию групп партий и расписаний их обработки в системе.

Цель и постановка задач. Цель выполняемой работы состоит в совершенствовании методов управления выполнением конвейеризированных программ обработки данных в многостадийных системах при наличии ограничений на используемые для этого ресурсы. Достижение сформулированной цели должно обеспечиваться решением задач: выполнения системного анализа поставленной проблемы, определения подходов к ее решению (способов ее решения), обоснования используемого для получения решения математического аппарата (выбор методов построения решения).

Основное содержание работы

Для формализации решения сформулированных задач в рассмотрение введены обозначения, соответствующие требованиям, обрабатываемым в системе. Т.к. понятию выполняемой в системе конвейеризированной программы, обрабатывающей данные, в теории расписаний соответствует понятие обрабатываемого требования, тогда в рассмотрение введены обозначения: N – множество типов требований ($N = \{1, 2, \dots, n\}$, n – количество типов требований), тип требования i соответствует наименованию (номеру) выполняемой в системе конвейеризированной программы, обработка которых реализуется в системе; N^i – множество, элемент n^i которого – это количество требований i -го типа, обработка которых должна быть выполнена ($i = \overline{1, n}$), d – момент времени поступления в конвейерную систему пакета заданий на обработку, т.е. время всех поступлений $\sum_{i=1}^n n_i$ требований n типов (начало пакетной обработки – в момент времени $d = 0$), d_i – момент времени поступления в систему на обработку требований каждого i -го типа (все программы, выполнение которых реализуется в системе, поступают в нее одновременно, тогда $d_i = 0$). Обработка требования i -го типа на l -ом приборе соответствует запуску на выполнение l -ой части i -ой конвейеризированной программы, которая находится в оперативной памяти l -го обрабатывающего прибора (все программы распределены по приборам и их соответствующие части находятся в оперативной памяти этих приборов). Трудоемкость обработки требования i -го типа на каждом l -ом приборе характеризуется параметром длительности его обработки на этом приборе, обозначенной t_i^l . Количество данных, хранение которых должно быть обеспечено в оперативной памяти l -го прибора для реализации обработки требования i -го типа обозначено через v_i^l . Значение v_i^l

измеряется в количестве ячеек памяти, которые должны быть выделены для хранения данных (в количестве единиц измерения ресурса $e = 2$). Для задания характеристик типов требований в рассмотрение также введены обозначения параметров: t_i^{nl} – интервал времени первоначальной наладки l -го прибора на обработку требований i -го типа, t_{jl}^{peri} – интервал времени переналадки с обработки требований i -го типа на обработку требований j -го типа l -го прибора. Также обрабатываемым в системе требованиям i -го типа соответствует маршрут их перемещения между приборами системы, обозначенный через tm_i (технологический маршрут для всех i -ых типов требований ($i = \overline{1, n}$) включает все приборы конвейерной системы, которые упорядочены в соответствии с их номерами), и дисциплина обслуживания требований в системе, обозначенная как ds_i (дисциплина обслуживания требований всех типов предполагает последовательное прохождение ими всех приборов системы). Т.к. приоритеты типов требований являются одинаковыми, то при построении решений они могут не учитываться. Тогда i -ый тип требований может быть сопоставлен с группой параметров вида:

$$\langle T_i, V_i, tm_i, ds_i, T_i^H, T_{jl}^{peri} \rangle, \quad (1)$$

где T_i – вектор длительностей обработки требований i -го типа на l -ых приборах системы (количества ресурса $e = 1$, задействованного на каждом l -ом приборе при обработке требований i -го типа, $l = \overline{1, L}$), V_i – вектор количества единиц ресурса $e = 2$, задействованных на каждом l -ом приборе ($l = \overline{1, L}$) при обработке требований i -го типа; T_i^H – вектор значений t_i^{nl} ; T_{jl}^{peri} – матрица значений t_{jl}^{peri} , (j, l) –ый элемент которой равен длительности интервала времени переналадки с обработки требований i -го типа на обработку j -го типа для l -го прибора (элемент $t_{il}^{peri} = 0$). Если следовать введенным обозначениям, то пакет заданий на обработку требований в вычислительной системе может быть представлен в виде множества наборов параметров вида $\langle i, N^i \rangle$, где типам требований i ($i = \overline{1, n}$) соответствуют наборы параметров (1) (при условии, что значения параметров $d_i = 0$).

Для того, чтобы определить количество e -ых ($e = \overline{1, 2}$) ресурсов системы, выделяемых для обработки требований, в рассмотрение введены следующие обозначения: z – индекс (номер) интервала времени, в течение которого реализуется обработка

требований (ресурс $e = 1$); Z – количество интервалов времени, в течение которых системой реализуется обработка требований; t^z – интервал времени функционирования системы при обработке требований ($z = \overline{1, Z}$); V^1 – объем (количество) ресурса $e = 2$ на l -ом обрабатывающем приборе системы, который выделяется при обработке требований i -ых типов ($i = \overline{1, n}$). Обработка требований может реализовываться в течение Z интервалов t^z ; при этом для обработки требований на l -м приборе выделяется V^1 единиц ресурса $e = 2$. В соответствии с введенными ограничениями на длительность интервалов t^z должны быть сформированы группы партий, каждая из которых обрабатывается в течение интервала t^z . Таким образом, группа партий соответствует их (партий) совокупности, обрабатываемой в течение интервала t^z . Выделяемые для обработки требований ресурсы системы могут быть определены набором параметров вида:

$$\langle Z, \{t^z | z = \overline{1, Z}\}, \{V^1 | l = \overline{1, L}\} \rangle. \quad (2)$$

После введения в рассмотрение наборов параметров (1) и (2) данные, поступающие на вход системы для формирования расписаний обработки требований i -х типов ($i = \overline{1, n}$) при наличии ограничений на выделяемые для этой обработки ресурсы, могут быть представлены в виде:

$$[\langle i, N^i \rangle | i = \overline{1, n}\rangle, \langle Z, \{t^z | z = \overline{1, Z}\}, \{V^1 | l = \overline{1, L}\} \rangle].$$

Таким образом, особенностями постановки задачи, подход к решению которой обосновывается в предлагаемой работе, являются: 1) необходимость выполнения программ каждого типа в конвейерной системе заданное число раз (необходимость обработки требований i -ых типов (при $i = \overline{1, n}$) заданное количество раз); 2) реализация обработки данных в системе осуществляется в течение заданного количества Z равных интервалов времени t^z (вводится ограничение на время функционирования системы обработки данных); 3) выполнение программ в конвейерной системе осуществляется при учете введенных ограничений на объем V^1 ($l = \overline{1, L}$) предоставленной для хранения обрабатываемых данных в оперативной памяти на каждом l -ом приборе. С точки зрения теории расписаний рассматриваемая постановка задачи предполагает необходимость обработки заданное (n^i) число раз требований каждого i -го типа ($i = \overline{1, n}$) при наличии ограничений на выделяемые для обработки этих требований ресурсы системы.

Обобщенная цель функционирования системы определяет необходимость обработки максимального количества требований заданных типов при ограничениях на используемые ресурсы. Для сформулированной цели функционирования системы необходимо выполнить ее декомпозицию [2] на подцели, каждая из которых реализуется определенным уровнем иерархии подсистем.

Результатом выполненной декомпозиции цели системы является определение функций, выполняемых иерархически упорядоченными подсистемами:

1) первый уровень выполняет формирование партий требований различных типов, эффективных по составу;

2) второй уровень выполняет определение состава групп партий, обрабатываемых в течение интервалов времени t^z ($z = \overline{1, Z}$);

3) третий уровень выполняет формирование расписаний обработки партий.

В соответствии с иерархическим упорядочением подцелей формулируются свойства процедуры принятия решений, предполагающие [2]:

1) порядок принятия решений, предполагающий первоочередное формирование решений подсистемами вышестоящих уровней;

2) зависимость эффективности решений на вышестоящих уровнях иерархии от решений нижестоящих подсистем. Для определения видов решений на уровнях иерархии в рассмотрение введены обозначения:

1) если через i обозначен тип требований, обработка которых выполняется в системе, тогда через m_i обозначим количество партий требований этого типа, формируемых на первом уровне иерархии подсистем, при $i = \overline{1, n}$ элементы m_i образуют вектор \mathbf{M} ; в рассмотрение вводится матрица \mathbf{A} , элемент a_{iu} которой соответствует количеству требований i -го типа в u -ой партии ($u \leq m_i$); матрица \mathbf{A} – это матрица составов партий требований i -ых типов;

2) решение, формируемое на верхнем уровне (количество и состав партии требований) имеет вид: $[\mathbf{M}, \mathbf{A}]$, где \mathbf{M} – вектор количества партий требований i -х типов, \mathbf{A} – матрица количества требований в партиях;

3) через N^z ($z = \overline{1, Z}$) обозначены группы партий, обрабатываемые в течение интервалов t^z ($z = \overline{1, Z}$); при распределении совокупности партий требований i -ых типов ($i = \overline{1, n}$), представленной в виде решения $[\mathbf{M}, \mathbf{A}]$, по группам партий N^z ($z = \overline{1, Z}$) состав партий не меняется; партии требований i -го типа могут входить в различные группы партий N^z ;

4) через m_i^z обозначено количество партий требований i -го типа в группе партий N^z (если партии требований i -го типа входят в N^z), через $(\mathbf{A})_i^z$ обозначим вектор количества требований i -го типа в m_i^z партиях в группе N^z ; для определения состава партий требований i -го типа, входящих в группу партий N^z , в рассмотрение введен набор параметров вида $[i, m_i^z, (\mathbf{A})_i^z]$, группа партий N^z – совокупность наборов, $N^z = \{[i, m_i^z, (\mathbf{A})_i^z]_k \mid k = \overline{1, k_z}\}$, либо $N^z = \{[i, m_i, (\mathbf{A})_i]_k \mid k = \overline{1, k_z}\}^z$; решение, формируемое на втором уровне иерархии подсистем – совокупность групп партий, – имеет вид: $\{N^z \mid (z = \overline{1, Z})\}$.

В оперативной памяти обрабатывающих приборов реализуется размещение как самих l -ых частей i -ых выполняемых программ, так и их данных (выполнение обработки данных обеспечивается размещением в оперативной памяти приборов всех обрабатывающих программ и обрабатываемых ими данных). Так как все l -ые части i -х выполняемых программ находятся в блоках оперативной памяти соответствующих l -ых приборов одновременно, то активизация их выполнения может быть выполнена в некотором порядке, эффективном с точки зрения задаваемого критерия. В свою очередь порядок запуска программ на обработку на каждом l -м приборе соответствует последовательности выполнения этих программ на этом приборе (последовательности активизации программ, находящихся в оперативной памяти). Совокупность последовательностей запуска программ на обработку на всех l -х приборах ($l = \overline{1, L}$) образует расписание выполнения программ при обработке данных в конвейерной вычислительной системе. В терминах теории расписаний последовательности (порядку) активизации частей программ находящихся в блоках оперативной памяти каждого обрабатывающего прибора соответствует последовательность (порядок) запуска требований соответствующего типа на обработку. В силу того, что обработке подлежат требования одинаковых типов, образующие партии, то расписанию обработки партий требований различных типов соответствует совокупность последовательностей (порядков) запуска на обработку партий требований.

Решение задачи эффективной обработки партий требований связано с построением расписаний (определением порядка обработки партий на приборах системы), которое выполняется при наличии ограничений на количество используемых ресурсов. Способ представления последовательностей обработки партий группы N^z (группа партий для соот-

ветствующего интервала t^z) предполагает введение в рассмотрение матрицы порядка обработки партий в системе $(P)^z$. Порядок обработки партий требований на приборах соответствует последовательности обработки этих партий, тогда обозначим через $\pi^1 (1 = \overline{1, L})$ последовательность обработки партий на l -ом приборе, которой соответствует матрица $(P)^z$. Таким образом, $\pi^1 (1 = \overline{1, L})$ – это обозначение последовательности (порядка) обработки партий на l -ом приборе, которая задается матрицей порядка $(P)^z$. Так как порядок обработки партий на всех приборах одинаков (последовательности $\pi^1 (1 = \overline{1, L})$ обработки партий на приборах одинаковы), то достаточно определения одной матрицы порядка $(P)^z$ для всей системы, элемент $p_{ij} = 1$, если партия требований i -го типа занимает j -ю позицию в π^1 , $p_{ij} = 0$ в противном случае. В рассмотрение вводится матрица $(R)^z$ – матрица количества требований в соответствующих партиях требований i -го типа, занимающих j -е позиции в π^1 (элемент r_{ij} равен количеству требований i -го типа в партии, занимающей j -ю позицию в последовательности π^1). В силу введенных обозначений решение, формируемое на третьем уровне иерархии подсистем (уровень определения порядков запуска на обработку партий требований различных типов), – расписание обработки партий требований на приборах системы имеет вид: $[(P)^z, (R)^z]$, при $z = \overline{1, Z}$. В рассмотренной постановке задача построения расписаний групповой обработки партий при наличии ограничений на ресурсы системы решается с использованием аппарата многоуровневого программирования. Обобщенная модель задачи многоуровневого программирования имеет вид ([3],[4]):

1-й уровень: $f_1(\alpha, \beta^*) \rightarrow \min, \alpha \in A$ при $q_1(\alpha) \leq a$,

2-й уровень: $f_2(\alpha, \beta, \gamma^*) \rightarrow \min, \beta \in B$ при $q_2(\beta) \leq b$,

3-й уровень: $f_3(\beta, \gamma) \rightarrow \min, \gamma \in \Gamma$ при $q_3(\gamma) \leq g$,

где α, β, γ – решения на первом, втором, третьем уровнях, A, B, Γ – множества решений на уровнях, $q_p | p = \overline{1, 3}$ – функции, с использованием которых задаются ограничения при выборе решений на этих уровнях, β^*, γ^* – эффективные решения, полученные на втором и третьем уровнях соответственно.

Процесс принятия эффективных решений в системе для рассматриваемой постановки задачи

должен выполняться с использованием совокупности критериев эффективности на каждом уровне иерархии подсистем. Переход к векторному критерию должен быть выполнен для подсистем, принятие решений в которых связано с непосредственным функционированием обрабатывающей требования системы. При этом необходимо различать внешние и внутренние цели функционирования системы: внешние цели обуславливают необходимость обработки требований, внутренние цели – необходимость эффективного использования оборудования.

Выполнен анализ внутренних и внешних целей применительно к определенным в результате декомпозиции уровням системы анализ:

1) функционирование первого уровня связано с обеспечением достижения внешней цели системы – обработки требований, поэтому эффективность решения на этом уровне оценивается на основе полученного решения со второго уровня (т.е. по размещенным в группах N^z партиям требований, состав которых формируется на первом уровне);

2) функционирование второго уровня обеспечивает достижение внешней (обработка требований) и внутренней (эффективность использования ресурсов системы (в частности, обрабатывающих приборов)) целей системы, следствием является формирование векторного критерия на втором уровне;

3) третий уровень, реализующий построение расписаний обработки партий требований, входящих в группы $N^{z*} (z = \overline{1, Z})$, должен обеспечивать формирование решений таким образом, чтобы обеспечить эффективное использование e -ых ресурсов системы ($e = \overline{1, 2}$), в рассмотрение должен быть введен векторный критерий, каждая из компонент которого (локальный критерий) определяет эффективность решения с точки зрения использования ресурсов системы при его (решении) реализации.

Тогда многокритериальная многоуровневая модель принятия решений при обработке групп партий имеет следующий обобщенный вид:

1) первый уровень:

$$f_1(\{N^{z*} | z = \overline{1, Z}\}) \rightarrow \min,$$

2) второй уровень:

$$\left(\begin{array}{l} f_{21}(\{N^z\}, \{[(P)^z, (R)^z]^*\}) \\ f_{22}(\{N^z\}, \{[(P)^z, (R)^z]^*\}) \end{array} \right) \rightarrow \min, \\ z = \overline{1, Z}, N^z \in B,$$

3) третий уровень:

$$\left(\begin{array}{l} f_{31}(\{N^z\}, \{[(P)^z, (R)^z]\}) \\ f_{32}(\{N^z\}, \{[(P)^z, (R)^z]\}) \end{array} \right) \rightarrow \min, \\ z = \overline{1, Z}, [(P)^z, (R)^z] \in \Gamma.$$

Выводы

В работе обоснован подход к построению многокритериальной модели многоуровневого программирования, дальнейшим развитием которого является разработка самой модели и метода построения эффективных расписаний обработки партий требований заданных типов.

Литература

1. Ковалев, М.М. *Модели и методы календарно-го планирования. Курс лекций. [Текст] / М.М. Ковалев. – Минск: Изд-во БГУ, 2004. – 63 с.*
2. Месарович, М. *Теория иерархических многоуровневых систем. [Текст] / М. Месарович, Д. Мако, И. Такахара. – М.: Мир, 1973. – 344 с.*
3. Петросян, Л.А. *Теория игр. [Текст] / Л.А. Петросян, Н.А. Зенкевич, Е.А. Семина. – М.: Высш. шк., 1999. – 300 с.*
4. Гермейер, Ю.Б. *Игры с противоположными интересами. [Текст] / Ю.Б. Гермейер. – М.: Наука, 1976. – 327 с.*

Поступила в редакцию 25.02.2013, рассмотрена на редколлегии 20.03.2013

Рецензент: д-р техн. наук, проф., проф. каф. автоматизации проектирования вычислительной техники Г.Ф. Кривуля, Харьковский национальный университет радиоэлектроники, Харьков, Украина.

ОБҐРУНТУВАННЯ БАГАТОКРИТЕРІАЛЬНОЇ БАГАТОРІВНЕВОЇ МОДЕЛІ БУДОВИ РОЗКЛАДІВ ГРУПОВОЇ ОБРОБКИ ПАРТІЙ ВИМОГ

К.В. Кротов, А.В. Скатков

Для постановки задач побудови розкладів обробки вимог в багатостадійних конвеєрних системах необхідно формування інформаційної моделі відповідної обробної системи, в якій формалізуються види вхідних даних, види обмежень на безліч рішень задачі, ресурси, які виділяються для обробки вимог. У роботі вводяться поняття, необхідні для формування інформаційної моделі конвеєрної системи, обґрунтовується необхідність переходу до ієрархічної задачі побудови розкладів (використання апарату ієрархічного програмування). Обґрунтовується у загальному вигляді багатокритеріальна модель багаторівневого програмування для формування ефективного складу груп партій вимог і розкладів обробки партій при наявності обмежень на використовувані ресурси обчислювальної системи.

Ключові слова: багаторівневе програмування, партії вимог, групи партій, комплексні розклади, багатостадійна система, обмеження на використовувані ресурси.

RATIONALE FOR A MULTI-CRITERIA MULTI-LEVEL MODEL OF THE STRUCTURE OF SCHEDULES OF GROUP PROCESSING THE REQUIREMENTS OF THE PARTIES

K. V. Krotov, A. V. Skatkov

For the staging of the challenges of building the schedule of processing requests in multiphase conveyor systems should be created information model of the corresponding manufacturing system, which formalizes the types of input data, the kinds of restrictions on the set of solutions, allocated to the claims processing resources. Notations for the formalization of information model of the conveyor system, the necessity of transition to a hierarchical task of building schedules (using the hierarchical programming) is grounded. It is justified in the general form of a multicriteria model of multi-level programming for the formation of the effective composition of the groups of parties of the requirements and schedule processing of parties in the on-existence of restrictions on the use of resources of a computer system.

Key words: multi-level programming, party requirements, the group of parties, the comprehensive timetable, multi-stage system, restrictions on the use of resources.

Кротов Кирилл Викторович – канд. техн. наук, доцент кафедры информационных систем Севастопольского национального технического университета, Севастополь, Украина, e-mail: krotov_k1@mail.ru,

Скатков Александр Владимирович – д-р техн. наук, профессор, заведующий кафедрой кибернетики и вычислительной техники Севастопольского национального технического университета, Севастополь, Украина, e-mail: kv.sevntu@gmail.com.