ВВЕДЕНИЕ

Актуальность темы. На современном этапе развития направления теории расписаний и задач оптимизации рассматриваются как приближенные, так и точные методы. В информационных технологиях часто появляется необходимость обработки больших массивов разнотипных данных за ограниченное время. Одним из способов решения данной проблемы является применение систем конвейерного типа. В этих системах важной составляющей является поступление требований на обработку. Теория расписаний является частью исследования операций. Теория расписаний исследует задачи, в которых необходимо упорядочить или, другими словами, определить последовательность выполнения совокупности работ, использования каких-либо средств и т.д.

Задачи упорядочения носят самый общий характер. Они возникают там, где существует возможность выбора той или иной очередности выполнения работ: при распределении работ на производстве, составлении расписания приземления самолетов, составлении расписания движения поездов, обслуживании клиентов в обслуживающих системах и т.д.

Результаты, к которым приводит то или иное упорядочение, существенно отличаются. В ряде практических случаев эти различия принимают стоимостный характер или определяются какой-либо другой величиной.

Цель и задачи работы. Целью выпускной квалификационной работы является разработка методов построения решения по оптимизации составов партий данных и расписаний их обработки. Для решения поставленной задачи была введена трехуровневая иерархическая модель. На основании полученной модели были поставлены задачи:

формирование метода оптимизации составов партий всех типов данных;

формирование метода оптимизации составов партий фиксированного типа данных;

формирование метода оптимизации расписаний обработки партий данных при наличии ограничения на время функционирования системы.

Предмет и объект исследования. В выпускной квалификационной работе рассматривается многоуровневая система теории расписания. Теория расписаний ‑ раздел дискретной математики, занимающийся проблемами упорядочения. В большинстве случаев в теории расписания ставится задача дискретной оптимизации: построить расписание, минимизирующее время выполнения работ, стоимость работ и т.п. Расписание ‑ указание, на каких машинах и в какое время должны обслуживаться требования (выполняться работы).

Производительность обработки данных при выполнении программ можно повысить путём конвейеризации [1]. Под конвейеризацией выполнения программы подразумевается разделение её на фрагменты, каждый из которых закреплён за соответствующим сегментом конвейера. Выполнение программ реализуется в многостадийной последовательной обрабатывающей системе с заданным порядком сегментов конвейера. Сегменты конвейера следуют строго друг за другом, местоположение сегмента в последовательности определяется его номером. Обозначим количество сегментов в конвейере через *L*, тогда порядковый номер сегмента, входящего в состав конвейера - . Если в состав конвейера входит *L* сегментов, то все выполняемые в системе программы должны быть разделены на *L* фрагментов, каждый из которых закреплён для выполнения за соответствующим сегментом конвейера. Тогда выполнение конвейеризированной программы в составе конвейера предполагает строгое закрепление её *l*-того фрагмента для выполнения на *l*-том сегменте конвейера.

В систему обработки данных могут поступать различные типы данных, обозначим их количество через *n*. Через *i* обозначим номер множества однотипных данных, которые должны быть обработаны в системе, тогда . Количество элементов в множестве однотипных данных, характеризуемых индексом *i*, обозначим через . Все данные *i*-того типа (в количестве ) обрабатываются соответствующей им программой. То есть существует типов программ, выполняемых в системе, обрабатывающих данные *i* типа. Таким образом, в системе задаётся строгое соответствие между типом данных и обрабатывающей эти данные программой. Однократное выполнение программы *i*-го типа позволяет обработать один элемент множества данных *i*-того типа. Так как множество *i*-того типа содержит элементов, то обрабатывающая программа *i*-того типа должна быть выполнена число раз. Цель функционирования системы в этом случае состоит в обработке всех поступающих на её вход данных. В конвейеризированной системе, которая состоит из *L* сегментов, реализуется обработка n типов данных n типами программ. Для повышения эффективности использования ресурсов системы рекомендуется обрабатывать данные партиями [2, 3]. В этом случае «партия» - это совокупность наборов данных одного (*i*-того, ) типа, которые обрабатываются в программе непосредственно друг за другом.

Выполнение программ в конвейеризированной вычислительной системе требует использования её (системы) ресурсов. В первую очередь такими ресурсами являются процессорное время каждого сегмента конвейера и оперативная память, в которой хранится выполняемая программа и обрабатываемые данные. Чтобы наиболее эффективно с точки зрения использования сегментов конвейера использовать вычислительную систему необходимо планировать запуск программ различных типов, то есть составлять расписания обработки данных разных типов. Так как поступление данных *i*-того типа инициирует в конвейеризованной системе выполнение программы *i*-того типа, то под «расписанием» понимается порядок поступления данных разных типов на вход системы или, другими словами, порядок запуска программ разных типов на выполнение.

Научна новизна. В выпускной квалификационной работе рассматривается градиентный подход с использованием жадной стратегии для решения задачи о оптимизации составов партий. Градиентный метод использует понятие окрестности и шага решения. Под окрестностью понимают набор возможных решений для получения локального максимума в текущей окрестности. Под локальным максимумом понимается понятие наилучшего решения среди всех из текущей окрестности. Под шагом алгоритма понимается понятие увеличения окрестности и получения большего числа потенциальных локальных максимумов для нахождения среди них глобального оптимального (наилучшего решения для поставленной задачи).

Практическое значение работы. Рассматриваемая система применима в конвейерном производстве. Система позволяет оптимизировать составы партий данных для их обработки за фиксированное время работы. Система построена таким образом, что может использоваться в любых отраслях, применяющих конвейерную обработку данных. В текущем контексте разработанная программа доказывает необходимость использования полученной системы по сравнению с системой, основанной на фиксированных партиях (составы партий не изменяются). В дальнейшем планируется разработать программный продукт, основанный на полученной системе, конвейерной обработки спутниковых снимков земли (эта задача актуальна, так как обработка большого количества изображений занимает огромное время).

Структура работы. Выпускная квалификационная работа состоит из введения, семи основных разделов, заключения, списка использованных источников и четырех приложений.

В первом разделе обоснована актуальность разработки методов оптимизации составов партий данных.

Во втором разделе представлена модель системы и методы оптимизации составов партий данных.

В третьем разделе описан метод построения и оптимизации порядка обработки партий данных.

В четвертом разделе описана программная реализация методов, описанных в разделе два.

В пятом разделе проводится анализ результатов работы разработанных методов оптимизации.

В шестом разделе представлено экономическое обоснование проектного решения.

В седьмом разделе представлены показатели охраны труда и методы их улучшения.

В результате выполнения выпускной квалификационной работы было разработано Windows-приложение, которое строит расписания обработки данных многих типов, с учетом времени перенастройки оборудования с обработки требования одного типа на другой. Разработанная система производит оптимизацию расписания по нескольким критериям эффективности, программный комплекс условно разделён на три уровня. Каждый из них представляет собой поиск оптимума по одному или нескольким критериям.

Из-за того что обработка на всех устройствах происходит последовательно, и обработка каждой партии не может быть разбита на несколько частей, необходимо сформировать порядок поступления партий данных только на первое устройство.

ПОСТАНОВКА ЗАДАЧИ

В конвейеризированной системе, которая состоит из L сегментов, реализуется обработка n типов данных n типами программ. Для повышения эффективности использования ресурсов системы рекомендуется обрабатывать данные партиями [2, 3]. В этом случае «партия» - это совокупность наборов данных одного (i-того, ) типа, которые обрабатываются в программе непосредственно друг за другом.

Входными данными для системы построения расписаний обработки данных разных типов при формировании комплектов являются:

- количество типов данных (n);

- количество элементов в множестве данных каждого типа (, );

- количество сегментов системы (L);

- матрица состава комплектов (W);

- длительность обработки данных i-того типа l-ым сегментом системы (фрагментом i-той программы);

- интервалы времени переналадки приборов с обработки данных одного типа на обработку данных другого типа.

В процессе функционирования системы необходимо определить приближенно эффективное (с точки зрения вводимых в рассмотрение критериев) количество и составы партий данных, эффективный с точки зрения формирования комплектов а так же эффективное (с точки зрения минимального времени формирования комплектов) расписание обработки партий данных разных типов. Данная задача является сложной, поэтому требуется вертикальная декомпозиция целей, в результате которой задача будет разбита на подзадачи.

В данной работе рассматриваются только два аспекта работы системы: формирование партий и формирование комплектов. Необходимо разработать метод получения оптимального количества и состава партий данных каждого типа и метод расчета количества получившихся комплектов.

Для достижения поставленной цели необходимо решить следующие задачи:

- проанализировать существующие методы построения расписаний обработки партий данных разных типов;

- сформулировать подход к построению расписаний обработки партий при наличии ограничений на директивные сроки выпуска комплектов;

- выбрать математический аппарат;

- построить математическую модель системы;

- разработать метод формирования эффективного количества и состава партий каждого типа данных, обрабатываемых в системе;

- разработать эффективный метод формирования составов комплектов;

- исследовать разработанные методы и сделать выводы об эффективности разработанных алгоритмов и областях возможного применения системы в целом.

Описание предметной области

При реализации мониторинга земной поверхности с использованием данных ДЗЗ возникает необходимость определения значений параметров, характекризующих условия развития и распространения природных явлений и техногенных воздействий на ОС, наличие которых было определено. Идентификация условий распространения природных явлений и техногенных воздействий на ОС призвана обеспечить возможность прогнозирования их развития в течение времени. При этом параметры разных видов, соответствующие условиям распространения и используемые при прогнозировании развития природных явлений и техногенных воздействий на ОС, определяются на основе данных ДЗЗ, поступающих от разных спутников. Тогда необходимым является комплексирование результатов обработки спутниковых данных при определении условий распространения природных явлений и техногенных воздействий. При условии, что в буфере приема накапливаются данные от разных спутников, конвейерная система реализует обработку данных различных типов, соответствующих условиям распространения. Также на обработку поступает не единственный спутниковый снимок определенного типа, а их набор. В результате в буфере хранения данных системы мониторинга природных явлений и техногенных воздействий на ОС одновременно находятся несколько наборов данных разных типов (несколько наборов снимков от различных спутников, характеризующих условия распространения (развития) различных явлений и техногенных воздействий). Тогда из результатов обработки в конвейерной системе разнотипных спутниковых данных формируются комплекты различных составов.

Таким образом, решение задачи планирования обработки различных наборов спутниковых снимков с целью формирования комплектов значений параметров из результатов обработки связано с необходимостью формирования партий данных разных типов.

Каждый комплект параметров (результатов обработки данных ДЗЗ разных типов) характеризует определенное природной явление и техногенное воздействие, которому он соответствует. То есть каждый комплект представляет собой набор значений параметров, полученных в результате обработки спутниковых данных разных типов, соответствующих условиям распространения (развития) определенного природного явления и техногенного воздействия в соответствующем регионе. Тогда комплект – это набор значений параметров, являющихся результатами обработки спутниковых данных разных типов, представляющих собой условия, необходимые для прогнозирования развития природного явления либо техногенного воздействия на ОС определенного типа.

В том случае, если в конвейерной системе выполняется обработка наборов спутниковых данных различных типов, тогда возникает задача управления процессом обработки этих данных. Управление процессом обработки данных должно быть реализовано таким образом, чтобы обеспечить выполнение требования оперативности выдачи результатов и, как следствие, требования оперативности формирования комплектов данных. Наборы данных разных типов могут быть обработаны в системе целиком либо могут быть разбиты на части, называемые партиями данных.

Решение задачи управления процессом обработки партий данных в конвейерной системе предполагает построение комплексных расписаний обработки этих партий. Задача построения расписаний обработки партий данных с учетом требования формирования комплектов из результатов обработки также является комплексной и предполагает решение следующих подзадач: определения составов партий данных, построения расписаний обработки сформированных партий, распределения результатов обработки данных в партиях по комплектам заданных составов.

Обработка партий данных с целью идентификации условий развития (распространения) природных явлений и техногенных воздействий на ОС осуществляется таким образом, чтобы гарантировать оперативность получения результатов, из которых будут сформированы комплекты (гарантировать оперативность формирования каждого комплекта из резельтатов обработки). Тогда решение задачи управления процессом обработки партий данных (планирования обработки партий данных) осуществляется в соответствии с условием наиболее быстрого формирования комплектов каждого типа из результатов этой обработки (с точки зрения минимизации моментов времени окончания формирования каждого комплекта каждого типа).

Альтернативным способом решения задачи управления обработкой партий данных является учет задаваемых для комплектов каждого типа директивных сроков. Тогда совместное решение сформулированных задач (определения составов партий, расписаний их обработки и формирования комплектов из результатов обработки) должно гарантировать получение комплектов в заданные для них директивные сроки. Таким образом, к некоторым задаваемым моментам времени должны быть сформированы определенные комплекты установленного состава.

Обоснование моделей процесса формирования комплектов из результатов обработки данных, группируемых в парии, оптимизации решений по составам партий и расписаниям их обработки с учетом формирования комплектов

С целью решения задачи управления процессом обработки партий данных выполнена декомпозиция обобщенной цели функционирования системы на совокупность иерархически упорядоченных целей подсистем (подфункций, решаемых на иерархически упорядоченных уровнях системы построения комплексных расписаний обработки партий данных). Формирование решений осуществляется на уровнях системы следующим образом: первый– решения по составам партий данных вида *[M,A]*, второй– решения по порядку обработки на сегментах конвейера сформированных партий в виде . При этом составы партий, сформированные на первом уровне, передаются на второй уровень для построения расписаний их обработки, а сформированные эффективные решения по расписаниям обработки партий передаются на первый уровень для формирования из результатов, получаемых при обработке партий, соответствующих комплектов. Т.е. результаты по мере их готовности при завершении обработки каждой из партий распределяются по комплектам. Тогда на первом уровне иерархии подсистем определения решений выполняется решение двух взаимосвязанных задач:

– определения составов партий (решений вида *[M,A]*);

– формирования комплектов из результатов обработки данных в партиях с учетом определенного на втором уровне иерархии эффективного решения по порядкам их (партий) обработки .

В соответствии с выполненными рассуждениями управление процессом обработки партий данных в конвейерной системе при условии формирования комплектов задаваемых составов предполагает: 1) определение составов партий данных, обрабатываемых в системе (решений вида *[M,A]*); 2) определение порядка обработки партий данных – расписаний обработки партий в виде ; 3) распределение результатов, полученных при обработке партий данных, по комплектам заданного состава. При этом метод построения расписаний обработки партий отличается от метода, изложенного в разделе 4, т.к. определяемый с его использованием порядок выполнения операций с партиями должен обеспечивать оперативное формирование комплектов. Т.е. метод построения расписаний обработки партий реализует определения такого порядка партий в последовательностях (), который позволяет реализовывать оперативное формирование комплектов.

Рассматриваемая задача является задачей с полной информацией – все параметры, характеризующие обрабатываемые данные (типы данных, ко­личество данных, длительности обработки данных различных типов, составы комплектов различных типов и т.д.) и функционирующую систему (количество сегментов, дисциплина обработки партий, длительности интервалов времени функционирования системы и т.д.) являются заданными.

Для формализации модели процесса обработки партий данных в конвейерной системе при условии формирования комплектов в рассмотрение введены следующие обозначения. Через *g* обозначен идентификатор типа комплекта, который формируется из результатов обработки данных в конвейерной системе, через – общее количество типов формируемых комплектов. Комплект результатов обработки данных представляет собой совокупность данных, при этом количество данных каждого *i*-го типа (), входящих в комплект одного (*g-*го) типа, является заданным. Данные всех *n* типов входят в определенном количестве в составы комплектов каждого типа. Через обозначено количество данных *i*-го типа, которые должны входить в один комплект *g*-го типа. В рассмотрение введена матрица (*W*)*,* значения элементов которой равны числу данных *i*-го типа, которые включаются в один комплект *g*-го типа. Таким образом, количество результатов обработки данных *i-*го типа, включаемых в комплект *g*-го типа, обозначенное через , соответствует *i-*ой компоненте одного комплекта *g*-го типа.

Через обозначим количество комплектов *g-*го типа, которые должны быть сформированы из результатов обрабатки данных *i-*ых типов () в соответствии с заданием. Т.е количество комплектов заданного типа соответствует количеству однотипных явлений на земной поверхности, которые являются обнаруженными к текущему моменту времени, и для которых должны быть сформированы наборы параметров, характеризующих условия их развития (распространения) одинаковых составов. Тогда количество данных каждого *i-*го типа (), которые будут обработаны в системе с целью идентифкации условий развития (распространения) природных явлений и техногенных воздействий, определяется следующим образом: . В итоге все поступившие на вход системы для обработки данные будут использованы при формировании комплектов разных типов. Также через обозначим индекс (номер) сформированного комплекта *g-*го типа, тогда .

Входными данными для системы построения расписаний обработки партий при формировании комплектов являются: типы *i* () данных, обрабатываемых в системе; количество данных каждого *i-*го типа, которые должны быть обработаны с целью последующего формирования из результатов комплектов заданных составов; количество типов формируемых комплектов ; количество комплектов каждого типа (); матрица *(W)*,элемент которой равен количеству данных *i*-го типа, которые должны входить в один формируемый комплект *g*-ого типа ( ), матрица () длительностей обработки данных *i-*ых типов на *l-*ых сегментах конвейера, матрицы () длительностей переналадок *l-*ых сегментов конвейера с обработки данных *i-*го типа на обработку данных *j-*го типа. Выходными решениями, обеспечивающими формирование комплектов с учетом вводимых в рассмотрение критериев являются: количество и составы партий данных *i*-ых типов (); расписания обработки партий данных в конвейерной системе.

В соответствии с функциями уровней системы между ними выполняется обмен информацией следующего вида: 1) на вход первого уровня подаются: количество *n* типов обрабатываемых данных, количество типов комплектов , которые должны быть сформированы из данных, обработанных в партиях, количество комплектов каждого *g*-го типа (), количество () данных каждого типа, обрабатываемых в системе (определяемое на основе заданных значений количества комплектов каждого *g*-го типа ()); матрица *(W)* составов комплектов; с выхода уровня передаются составы партий данных *i*-ых типов () – решение *[М, А]*; 2) на вход второго уровня – решение *[М, А]*; с выхода уровня – сформированное для решения *[М, А]* эффективное расписание обработки партий, на основе которого реализуется распределение результатов между комплектами по мере их (результатов) готовности при окончании обработки каждой из партий. Т.е. в силу требования оперативности идентифкации условий развития (распространения) природных явлений и техногенных воздействий, полученные результаты распределяются по комплектам по мере окончания обработки партий в системе. При формировании расписаний обработки партий данных *i-*ых типов состав партий не изменяется (значения и , поступившие с верхнего уровня, изменены быть не могут).

В соответствии с обозначениями для моментов времени начала обработки *q-*ых данных в партиях, занимающих в последовательностях обработки *j*-ые позиции, определено обозначение для момента времени окончания обработки на *l-*ом сегменте *q-*ых данных в партии, в *j-*й позиции в последовательности (). По аналогии вводится обозначение для моментов времени окончания обработки в системе (соответственно, на *L*-ом сегменте конвейера) требуемого количества данных *i-*го типа, включаемых в формируемый -ый комплект *g*-го типа (,) в виде: . Момент времени равен моменту времени окончания обработки партии, результаты из которой позволяют закончить формирование *i-*й компоненты -го комплекта *g*-го типа, характеризуемой значением .

Таким образом, – это момент времени окончания обработки в соответствующей партии данных *i*-го типа, результаты из которой включаются в *i-*ю компоненту -го комплекта *g-*го типа таким образом, чтобы их количество в этой компоненте соответствовало значению .

В силу сказанного для -го комплекта *g*-го типа (, ) определен вектор моментов времени окончания обработки всех данных *i-*ых типов (в требуемом их количестве при ), обозначенный в виде , формируемый из значений . Вектор является -ым столбцом матрицы , компонентами которой являются элементы . Т.е. элементами матрицы являются моменты времени окончания формирования *i-*ых компонент в количестве в -ых комплектах *g*-го типа (размерность матрицы ). Количество матриц характеризуется значением .

Таким образом, на верхнем уровне с использованием решения по порядкам обработки партий данных вида , сформированного для соответствующего решения по составам партий , определяются матрицы () значений моментов времени окончания формирования *i*-ых компонент в -ых комплектах *g*-ых типов ().

Для каждого -го комплекта *g*-го типа () вводится в рассмотрение параметр, обозначенный как , соотвествующий моменту времени окончания его формирования. Значения определяются на основе значений следующим образом: . Т.е. момент времени окончания формирования -го комплекта *g*-го типа определяется как максимум среди моментов времени окончания формирования необходимого количества данных в каждой его *i-*ой компоненте ().

Цель управления процессом обработки состоит в обеспечении оперативности получения результатов – условий развития (распространения) каждого природного явления и техногенного воздействия на ОС по мере обработки данных (по мере реализации процесса обработки партий данных), т.е. в оперативности формирования комплектов из результатов обработки партий по мере завершения выполнения в системе операций с ними. Таким образом в случае завершения выполнения обработки партии, получаемые результаты распределяются по комплектам, и для каждого комплекта определяется момент времени его формирования. Т.к. значения для каждого -ый комплекта *g*-го типа () вычисляются на основе значений элементов матриц , тогда эти матрицы характеризуют решение по распределению результатов по комплектам, формируемое на основе решения на втором (нижнем) уровне иерархии.

В общем виде модель иерархической игры для определения составов партий при условии формирования комплектов в соответствии с введенными обозначениями имеет следующий вид:

– первые уровень:

, (5.1)

– второй уровень:

(5.2)

Т.к. эффективность решений по составам партий определяется с точки зрения оперативности формирования комплектов, а формирование комплектов реализуется на основе решения по порядку обработки партий , тогда модель (5.1), (5.2) может быть представлена в следующем упрощенном виде:

– первые уровень:

, (5.3)

– второй уровень:

. (5.4)

Если для -го комплекта *g*-го типа данные всех *i-*ых типов в количестве () подготовлены (т.е. сформированы *i*-е компоненты комплекта), тогда этот комплект является сформированным в качестве совокупности параметров, характеризующих условия распространения (развития) соответствующего природного явления или техногенного воздействия на ОС. Оперативность формирования каждого -го комплекта *g*-го типа (,) характеризуется минимальным значением момента времени (при условии, что результаты из партий распределяются по комплектам по мере окончания их (этих партий) обработки в конвейерной системе).

Выполнение операций со всеми партиями данных *i-*ых типов () в конвейерной системе начинается в момент времени . Значение характеризует интервал времени обработки данных в партиях с целью формирования первого комплекта *g*-го типа (временные затраты на обработку данных в партиях, связанные с формированием первого комплекта *g*-го типа). Во второй комплект *g-*го типа могут быть включены данные из таких партий, для которых (т.е. результаты, полностью формирующие некоторую *i-*ю компоненту второго комплекта *g-*го типа, получены при обработке партии в *j*-ой позиции в (), выполнение операций с которой завершается до окончании формирования первого комплекта *g-*го типа). Таким образом, результаты обработки данных некоторого *i-*го типа в партии в *j-*ой позиции в () могут быть распределены по *i-*ым компонентам нескольких комплектов *g-*ого типа (либо *g-*ых типов, ). Поэтому интервал времени, затраченный на получение результатов, из которых формируется второй комплект *g-*го типа, не может быть определен выражением , а характеризуется значением, соответствующим . Аналогичные рассуждения могут быть применены для любого -го комплекта *g-*го типа. Тогда, временные затраты на обработку данных в партиях с целью формирования всех -ых комплектов () *g*-го типа определяются выражением . Оперативность формирования комплектов обеспечивается составами партий, для которых значение, получаемое с использованием приведенного выражения является минимальным. Тогда для комплектов каждого *g-*го типа () требуется обеспечить минимизацию значения времени, затрачиваемого на их формирование. В итоге по всем *g-*ым типам комплектов необходимо выполнить совместную минимизацию значений временных затрат, связанных с обработкой данных в партиях с целью получения результатов, из которых формируются комплекты. Тогда вид выражения, с использование которого реализуется определение суммарных временных затрат на формирование комплектов *g-*ых типов () следующий: . Данное выражение представляет собой аддитивную свертку ([180-183] выражений, определяющих временные затраты на формирование комплектов каждого *g-*го типа ().

Эффективность применения метода определения составов партий для задачи управления обработкой данных при формировании комплектов определяется путем сравнения получаемых решений с решением для фиксированных партий. В случае обработки фиксированных партий все комплекты формируются одновременно в момент времени окончания обработки последней партии. В этом случае полученное выражение не является приемлемым для определения числовой характеристики решения. Поэтому для идентификации характеристик решений предложен способ, позволяющий определить средние временные затраты на обработку партий данных, связанные с получением результатов, из которых формируются комплекты (средний интервал времени, затрачиваемый на обработку данных в партиях для получения результатов, из которых формируются комплекты). Выражение для определения средних временных затрат на выполнение операций с данными в партиях, из результатов обработки которых формируются комплекты, имеет вид . Предложенное выражение используется в качестве критерия для оптимизации решений по составам партий при реализации требования оперативности и равномерности формирования комплектов из результатов обработки.

На основе выполненных в главе 4 рассуждений построение расписаний обработки партий на нижнем уровне иерархии реализуется в точки зрения минимизации общего суммарного простоя сегментов конвейера при обработке всех сформированных партий. Для рассматриваемой задачи принятия решений по расписаниям обработки партий предложен метод определения порядка их обработки. Этот метод предусматривает добавление партий в последовательности () таким образом, чтобы обеспечить наиболее быстрое формирование комплектов всех типов. При этом порядок обработки партий в последовательностях оптимизируется с точки зрения минимизации суммарного времени простоя сегментов конвейера.

Тогда модель двухуровневой иерархической игры, позволяющей реализовать определение составов партий, гарантирующих выполнение требования оперативности формрования комплектов с условиями развития (распространения) природных явлений и техногенных воздействий на ОС, имеет следующий вид:

– первые уровень:

, ; (5.5)

– второй уровень:

, (5.6)

Таким образом, модель (5.5), (5.6) соответствует первой задаче оптимизации решений по составам партий с точки зрения оперативности и равномерности формирования комплектов из результатов обработки.

В отличие от рассмотренной выше вторая задача оптимизации составов партий с учетом формирования комплектов из результатов обработки предполагает задание директивных сроков для каждого -ого комплекта *g-*го типа (,). Тогда совместное решение сформулированных задач (определения составов партий, расписаний их обработки и формирования комплектов из результатов обработки) должно гарантировать получение комплектов в заданные для них директивные сроки.

Т.к. решаемая задача предполагает задание для каждого комплекта директивных сроков его формирования, тогда в рассмотрение введена матрица , элемент которой представляет собой задаваемый момент времени окончания формирования -го комплекта *g-*го типа (,). При этом для каждого *g-*го типа комплекта элементы матрицы упорядочены в ее *g-*ой строке по возрастанию значений моментов времени окончания формирования комплектов: , если . Тогда обработка данных в конвейерной системе направлена на осуществление операций на всех сегментах конвейера с партиями данных разных типов таким образом, чтобы формирование комплектов из результатов выполнялось в соответствии с задаваемыми директивными сроками.

Определение эффективных решений по составам партий данных *i-*ых типов () вместо фиксированных партий позволяет получить лучший результат с точки зрения формирования из результатов обработки комплектов в установленные сроки. При этом достигается внешняя цель функционирования системы – обеспечение в заданные сроки формирования из результатов обработки данных комплектов соответствующих типов.

Входными данными для системы построения расписаний обработки партий при формировании комплектов с учетом задаваемых для них директивных сроков являются: типы *i* () данных, обрабатываемых в системе; количество типов комплектов , формируемых из результатов обработки; количество комплектов каждого типа (); матрица *(W)*,элемент которой равен количеству данных *i*-го типа, которые должны входить в один формируемый комплект *g*-ого типа ( ); количество данных каждого *i*-го типа, которые обрабатываются в системе; матрица директивных сроков формирования комплектов, элемент которой соответствует задаваемому моменту времени окончания формирования -го комплекта *g-*го типа (,). Выходными решениями, обеспечивающими формирование комплектов в установленные директивные сроки, являются: количество и составы партий данных *i*-ых типов (); расписания обработки партий данных в конвейерной системе.

В соответствии с изложенным подходом к декомпозиции обобщенной функции системы на совокупность иерархически упорядоченных подфункций, формирование решений на уровнях системы построения расписаний обработки партий осуществляется следующим образом: первый уровень – решения по количеству и составам партий, а также распределение результатов по комплектам после определения расписаний обработки партий, второй уровень – решения по порядку обработки партий сформированного состава на сегментах конвейера. Выполненная декомпозиция обобщенной функции системы на совокупность иерархически упорядоченных подфункций позволяет рассмотреть задачу как задачу теории иерархических игр [109-114].

В соответствии с функциями уровней системы между ними выполняется обмен информацией следующего вида: 1) на вход первого уровня подаются: количество *n* типов обрабатываемых данных; количество () данных каждого типа, обрабатываемых в системе (определяемое на основе заданных значений количества комплектов каждого *g*-го типа ()); матрица *(W)* составов комплектов; матрица директивных сроков окончания формирования -ых комплектов *g-*ых типов (); параметры, определяющие временные характеристики длительностей обработки данных и длительностей переналадки сегментов конвейера на обработку данных каждого типа; с выхода уровня передаются составы партий данных *i*-ых типов () – решение *[М, А]*; 2) на вход второго уровня – решение *[М, А]*; с выхода уровня – сформированное для решения *[М, А]* эффективное расписание обработки партий, на основе которого реализуется распределение результатов обработки данных по комплектам заданных составов. В общем виде в соответствии с введенными обозначениями модель иерархической игры имеет форму, аналогичную (5.3), (5.4).

Комплекты формируются только из результатов обработки данных, которые входят в партии. Полное формирование одного комплекта *g*-го типа предполагает получение (в требуемом количестве) всех результатов обработки данных, для этого необходимых. Формирование результатов обработки данных (в требуемом количестве), являющихся компонентами комплекта, требуется реализовать к директивному сроку окончания его формирования. Поэтому наилучшим решением по составам партий будет являться решение (с учетом построенного расписания обработки партий), которое обеспечивает формирование комплектов разных типов в соответствии с их директивными сроками. При определении эффективных решений на первом уровне должно идентифицироваться соответствие сроков окончания формирования комплектов (для решений по составам партий и расписаниям их обработки) заданным директивным срокам. Данное соответствие характеризуется степенью превышения реальных сроков формирования комплектов заданным директивным срокам. Это условие соответствует внешней цели функционирования системы и интерпретируется при формировании критерия эффективности решений на верхнем уровне иерархии.

Решение рассматриваемой задачи планирования обработки партий при условии формирования комплектов из результатов и задании для них директивных сроков обеспечивается предварительным упорядочиванием комплетков различных *g-*ых типов в соответствии с их директивными сроками . Тогда для обеспечения решения этой задачи в рассмотрение введены обозначения, использование которых позволит сформулировать вид критерия оптимизации и алгоритм упорядочивания идентификаторов комплектов: – упорядоченное множество идентификаторов типов комплектов, формируемое в результате реализации формулируемого алгоритма (мощность множества равна суммарному количеству комплектов, формируемых из результатов обработки, т.е. , порядок элементов в этом множестве соответствует порядку формирования комплектов разных типов с точки зрения их директивных сроков); – количество элементов в множестве , т.е. ; – упорядоченное множество значений директивных сроков окончания формирования всех -ых комплектов () *g*-ых типов () (мощности множеств и равны и определяются выражением ); *h*– позиция *g-*го типа комплекта в множестве и момента времени  в множестве ;  – момент времени окончания формирования комплекта *g-*го типа, идентификатор которого занимает *h*-ю позицию в множестве ; – директивный срок окончания формирования комплекта *g-*го типа, занимающиего *h-*ю позицию в множестве ; – запаздывание с формированием комплекта *g-*го типа, занимающего *h*-ю позицию в множестве . После реализации алгоритма упорядочивания комплектов всех типов множество  имеет вид . Между элементами множеств и установлено однозначное соответствие, т.е. некоторыму *g*-му типу комплекта, занимающему в множестве определенную *h*-ю позицию, соответствует директивный срок его формирования (также в *h*-ой позиции в множестве ). Таким образом, множество  может быть представлено в виде :, множество  в виде: . Тогда, при реализации алгоритма распределения результатов по формируемым комплектам определяются моменты времени  окончания формирования комплектов *g-*ых типов, занимающих в множестве  *h*-е позиции, которым соответствуют директивные сроки , занимающие в множестве также *h*-е позиции.

В случае  фиксируется запаздывание с формированием комплекта *g*-го типа относительно его директивного срока  (элемент соответствует *g-*ому типу комплекта, занимающему *h-*ю позицию в множестве ; элемент занимают *h-*ю позицию в множестве ). Для комплекта *g*-го типа (в *h*-ой позиции в множестве ) запаздывание с его формированием определяется выражением вида: , которое обеспечивает вычисление требуемого значения в случае . Для определения вида критерия в рассмотрение введен параметр , соответствующий интервалу времени запаздывания с формированием комплекта *g*-го типа (в *h*-ой позиции в ) по сравнению с заданным для него директивным сроком . Т.е. значение  определяется выражением вида: . Тогда для всех комплектов *g-*ых типов в *h*-ых позициях в множестве  (обозначенных как ) общее запаздывание с их формированием будет определено выражением . Данное выражение используется в качестве критерия на верхнем уровне принятия решений по составам партий.

В силу выше сказанного, на нижнем уровне принятия решений по расписаниям обработки партий реализуется метод определения порядка обработки, который предусматривает добавление партий в последовательности  () таким образом, чтобы обеспечить наиболее быстрое формирование комплектов всех типов, и определение порядка выполнения операций с ними в последовательностях () с точки зрения минимизации суммарного времени простоя сегментов конвейера.

Тогда, двухуровневая модель иерархической игры для определения эффективных составов партий данных и расписаний их обработки при формировании комплектов и заданных для них директивных сроках имеет итоговый следующий вид:

– первый уровень:

, ; (5.7)

– второй уровень:

, (5.8)

.

Использование моделей (5.5), (5.6) и (5.7), (5.8) позволяет выполнить оптимизацию составов партий и расписаний их обработки с учетом требования оперативности формирования комплектов из результатов и соответствия моментов времени формирования комплектов заданным для них директивным срокам.

Для определения вида модели вычислительного процесса обработки партий данных введены следующие обозначения: – время обработки данных *i*-го типа на *l*-ом сегменте конвейера (); – время переналадки *l*-го сегмента с обработки данных *i*-го типа на обработку данных *k*-го типа; – время первоначальной наладки *l*-го сегмента на обработку данных *i*-го типа; – матрица длительностей переналадок сегментов конвейера для типов данных, партии которых обрабатываются в системе; – время начала обработки партии данных *i*-го типа, занимающей в последовательности *j*-ю позицию; – матрица моментов времени начала обработки партий данных *i*-ых типов, занимающих в  *j*-е позиции; – матрица моментов времени начала обработки *q-*ых данных в партиях, занимающих в  *j*-е позиции (*q* – порядковый номер данных в партии в *j-*ойпозиции в (, где – количество данных в партии, занимающей *j-*ю позицию в ). Элементы матрицы определяются следующим образом: , где , – момент времени начала обработки первых данных в партии, занимающей *j*-ю позицию в . Определение значений и (,; ; ) выполняется по аналогии с подходом [12].

Для (*q=1*)-ой позиции данных *i-*го типа в (*j=1*)-ой партии в выражение для определения формируется при учете параметра – времени наладки сегмента на обработку данных *i-*го типа: . Для *q>1* (,) имеем [1]: . В этом выражении первое слагаемое определяет длительность наладки сегмента конвейера на обработку данных в первой позиции в , второе слагаемое–длительность обработки данных в первой партии, предшествующих данным в *q-*ой позиции. Для позиции (*q=1*) данных (*j=2*)-ой партии в для имеем: если – время переналадки первого сегмента конвейера с обработки данных *i-*го типа (первая позиция партии в ) на обработку данных другого типа (вторая позиция партии в ), а – количество данных *i*-го типа в партии в первой (*j=1*) позиции в , тогда , где значение определяется следующим образом:

, где .

Для позиций () значения определяются выражением:

, .

Обобщенное выражение для при для последовательности имеет вид: , моменты времени начала обработки *q*-ых данных в партии с : . Выражения для и сформированы следующим образом [12]:

;

;

.

Полученные выражения представляют собой модель вычислительного процесса обработки партий данных *i-*ых типов () на *l-*ых сегментах конвейера (). В соответствии с введенными обозначениями вид решения, формируемого на нижнем уровне иерархии системы, примет следующую форму: .

Метод построения расписаний обработки партий данных на втором уровне системы реализует жадный подход к оптимизации. Он предполагает добавление текущей рассматриваемой партии в конец последовательностей (), в которых на предыдущих шагах алгоритма были размещены партии данных, и определение эффективного местоположения этой пар­тии в (определение для рассматриваемой партии позиций в (), которые обеспечивают локально оптимальное решение). Внутренней целью функционирования системы является полное использование временного ресурса сегментов конвейера. Построение решений по порядкам обработки партий на сегментах конвейера в соответствии с этой целью обеспечивает формирование «плотных» расписаний. В этом случае решение по порядку обработки партий должно быть построено таким образом, чтобы временной ресурс системы использовался в полной мере. Тогда критерий эффективности расписаний обработки партий соответствует внутренней цели функционирования системы, определяющей необходимость минимизации простоев оборудования при обработке. При определении текущей эффективной позиции рассматриваемой партии в значение критерия характеризует общие простои всех сегментов конвейера при обработке текущего количества партий, находящихся в (для всех партий, включенных в последовательности ()).

В этом случае критерий эффективности расписания обработки партий на нижнем уровне учитывает: а) время простоя сегментов в ожидании начала обработки партий данных (с учетом интервалов наладки, перена­ладки и последующего ожидания); б) время простоя сегментов в ожидании готовности данных при их обработке внутри партий. Простой *l*-го сегмента в ожидании начала обработки первой в партии равен значению , суммарное время простоя сегментов в ожидании начала обработки партий в последовательностях определяется выраже­нием . Интервал простоя *l*-го сегмента в ожидании начала обработки следующей партии после окончания обработки предыдущей определяется выражением вида: , где *j* >1, – количество данных в предшествующей (*j-1)-*ой партии. Суммарный простой *l*-го сегмента в ожидании начала обработки всех *j-*ых партий () определён следующем образом: . В этом случае суммарной простой всех *L* сегментов в ожидании начала обработки партий на них (простой между партиями для всех *L* сегментов) определяется выражением:

. (3)

Простой *l*-го сегмента в ожидании готовности к обработке данных, занимающих *q*-ю позицию в *j-*ой партии в последовательности , определяется выражением вида: , где . Это выражение соответствует интервалу между данными в *q*-ой и *(q-1)-*ой позициях в *j-*ой партии в . Тогда суммарный простой *l*-го прибора в ожидании готовности к обработке всех данных в *j-*ой партии в вычисляется с использованием выражения:

, (4)

где *q –* номер позиции данных в *j-*ой партии в , – число данных в этой *j-*ой партии, . На основе (4) общий простой *l*-го сегмента в ожидании готовности к обработке данных внутри всех партий в определяется выражением вида: . Суммарный простой всех *L* сегментов конвейера в ожидании готовности данных внутри партий вычисляется выражением:

. (5)

Критерий эффективности решения на нижнем уровне иерархии для текущего количества партий, добавленных в учитывает общий простой сегментов конвейера при определении порядка обработки этих партий, тогда при его формировании должны быть учтены: выражение и формулы (3) и (5).

В соответствии с введенным обозначением вида определено обозначение для момента времени окончания обработки на *l-*ом сегменте конвейера *q-*ых данных в партии, занимающей *j-*ю позицию в последовательности (), в виде и аналогичное обозначение для момента времени окончания обработки в системе (соответственно, на *L*-ом сегменте конвейера) требуемого количества данных *i-*го типа, включаемых в формируемый -ый комплект *g*-го типа (,) в следующие виде: . Тогда для -ый комплекта *g*-го типа (,) может быть определен вектор моментов времени окончания обработки всех данных *i-*ых типов (в требуемом их количестве при ), обозначенный в виде , формируемый из значений для этих типов. Таким образом, на верхнем уровне с использованием решения по порядкам обработки партий данных вида , сформированного для соответствующего решения по составам партий , определяются вектора (,) значений моментов времени окончания обработки требуемого количества данных *i-*ых типов . Также для каждого -ый комплекта *g*-го типа вводится в рассмотрение обозначение для момента времени окончания его формирования. Значения определяются следующим образом: , т.е. момент времени окончания формирования -ый комплекта *g*-го типа определяется как максимум среди моментов времени окончания обработки необходимого количества данных всех *i-*ых типов (). Тогда в случае фиксируется запаздывание с формированием -ый комплекта *g*-го типа относительно его директивного срока . Т.к. цель управления вычислительным процессом состоит в обеспечении формирования комплектов в соответствии с директивными сроками, тогда критерием эффективности решений по составам партий является суммарное запаздывание в формированием всех -ый комплектов *g*-го типов. Для вычисления суммарного запаздывания с формированием комплектов при обработке данных в рассмотрение введен параметр , соответствующий интервалу времени запаздывания с формированием -го комплекта *g*-го типа по сравнению с заданным для него директивным сроком. Определение значения для конкретного комплекта g-го типа выполняется следующим образом:

(6)

Тогда суммарное запаздывание с формированием -ых комплектов *g*-го типов, соответствующее критерию принятия решений по составам партий на верхнем уровне, определяется выражением вида: , где значение параметра определяется выражением (6).

Способ определения значений моментов времени окончания обработки требуемого количества данных *i-*го типа, включаемых в -ые комплекты *g*-го типов, рассмотрен ниже. Для обоснования способа вычисления значений моментов времени окончания обработки в системе (на *L*-ом сегменте конвейера) требуемого количества данных *i-*го типа, включаемых в -ые комплекты *g*-го типов (, ,) , в рассмотрение введены следующие обозначения: 1) – текущий рассматриваемый тип данных, которые из партий распределяются по комплектам в порядке возрастания их идентификаторов *g*; 2) – счетчик количества данных *i’*-го типа, включенных в составы всех формируемых комплектов различных типов; 3) – идентификатор комплекта, который формируется на текущей итерации алгоритма; 4) – упорядоченное множество идентификаторов *g* типов комплектов, формируемое в соответствии с видом матрицы (множество идентификаторов типов комплектов, позиция типа *g* комплекта в котором определяется в соответствии со значением параметра ); 5) – упорядоченное множество идентификаторов *g* типов комплектов, являющееся копией множества , используемое при реализации алгоритма; 6) – множество идентификаторов позиций партий данных *i-*го типа в последовательности , имеющее вид: ; 7) – рассматриваемая позиция партии данных *i’*-го типа в последовательности ; 8) – порядковый номер комплекта рассматриваемого -го типа.

Алгоритм определения моментов времени окончания обработки данных *i-*ых типов в соответствии с расписанием вида , используемых при формирования -ых комплектов *g*-ых типов, имеет следующий порядок шагов:

1) для *g-*ых типов комплектов () задание значения параметра равным 1 ();

2) задание индекса *i’* типа данных, которые будут распределяться по комплектам *g*-ых типов (): , где *I* – множество типов данных, партии которых обрабатываются в системе; ;

3) задание значения счетчика количества данных *i’*-го типа, добавленных в -ый комплект *g’*-го типа, равным 0 ();

4)задание идентификатора *g’* вида комплектов, в которые будут включаться данные *i’*-готипа: ; ;

5) определение позиции *j’* в последовательности партии *i’*-го типа, данные из которой будут включаться в -ый комплект *g’*-го типа: , где *h –* порядковый номер позиции *j* партии *i’*-го типа в множестве ; ; инициализация значения индекса (номера) данных *q* в партии в *j’*-ой позиции в: ;

6) если , тогда выполняется переход на шаг 7; если , тогда выполняется переход на шаг 8;

7) если , тогда ; ; выполняется переход на шаг 6; если , тогда переход на шаг 5;

8) определение значения: ; изменение значения номера формируемого комплекта *g’*-го типа: ;

9) проверка условия ; в случае его выполнения – переход на шаг 10; при выполнении условия задание идентификатора вида комплекта *g’*, в который будут включаться данные *i’*-го типа: ; ; задание значения счетчика количества данных *i’*-го типа, добавленных в -ый комплект *g*-го типа равным 0 (); реализуется переход на шаг 6;

10) проверка условия , при его выполнении реализуется переход на шаг 11; при выполнении условия осуществляется модификация идентификатора типа *i’*, данные которого распределяются по комплектам g-ых типов ():, ; реализуется переход на шаг 3;

11) останов алгоритма;

Реализация сформулированного алгоритма позволяет сформировать для каждого -го комплекта () *g*-го типа () вектор значений моментов времени окончания обработки требуемого количества данных *i-*ых типов (). С использованием значений реализуется определение – моментов времени окончания формирования каждого -ый комплекта *g*-го типа из результатов обработки данных *i*-ых типов в конвейерной системе и последующее вычисление значения критерия на верхнем уровне вида (6), характеризующего текущее решение по составам партий данных.

Таким образом, двухуровневая модель иерархической игры для определения эффективных составов партий данных и расписаний их обработки при формировании комплектов и заданных для них директивных сроках имеет итоговый следующий вид:

1) первый уровень: , где , а значения параметра определяются выражением (6);

2) второй уровень: , где

Использование двухуровневой модели иерархической игры, представленной в виде совокупности критериев эффективности решений на каждом уровне иерархии системы, позволяет определять локально оптимальные решения по составам партий данных разных типов и расписаниям их обработки на сегментах конвейера.

Анализ существующих методов формирования составов партий для цонвейерной обработки спутниковых данных

Работы по созданию систем мониторинга ОС с использованием ДДЗ ведуться в различных научных организациях (ИКИ РАН, ИВТ СО РАН, ИАПУ ДВО РАН). Результаты работ по реализации таких систем изложены в [1-9]. Управление процессом обработки данных в системах, рассмотренных в [1-3], предусматривает, что диспетчер заданий идентифицирует наличие свободных ресурсов (вычислительного ресурса, ресурса оперативной памяти (ОП) и т.д.) для каждого типа данных из очереди, формирует сценарии обработки, представляющие собой последовательности запуска требуемых процедур, передает сценарии в виде файлов конфигурации на определенные вычислительные устройства (ВУ). На каждом из устройств данные определенных типов обрабатываются независимо. Работы [4,5] обобщают введенные в [1-3] принципы построения систем и способы управления обработкой данных. Для каждого типа данных выделяются сервер-диспетчер, формирующий запросы на получение данных от центров их хранения и предоставления, сервер-хранилище данных, реализующий хранение и тематическую обработку снимков определенного типа в соответствии с заданным для этого устройства сценарием. При освобождении сервера-хранилища данных он будет выделяться для обработки новых данных соответствующего типа. Работа [6] содержит рассматрение модификации базовой технологии обработки, представленной в [1-5]. Модификация предусматривает, что при поступлении в систему новых данных диспетчер заданий не ожидает освобождения ВУ, а определяет из них наличие требуемого для обработки количества ресурсов. В случае идентификации такого устройства, ему передаются данные вместе со сценарием обработки. Управление процессом обработки предусматривает не только идентификацию ВУ с требуемым количеством ресурсов, но и реализацию равномерного распределения данных по устройствам (управляющая процедура предотвращает загрузку всех ВУ обработкой данных одного типа).

В соответствии с [7] планирование выполнения заданий, находящихся в очереди, предусматривает использование освободившихся устройств, подходящих (с точки зрения наличия в ОП требуемых процедур) для интерпретации соответствующего сценария. Планирование выполнения заданий в системе, рассмотренной в [8], предусматривает использование системы управления обработкой Taverna. В ней процесс обработки представляется в виде графа потока управления. Выполнение различных рабочих процессов, связанных с обработкой данных, распределяется по ВУ, в ОП которых имеются требуемые процедуры. При этом рабочему процессу для реализации обработки выделяется тот ресурс, который менее всего загружен выполнением других рабочих процессов. Подход к управлению процессом обработки данных, рассмотренный в [9], аналогичен способу управления, рассмотренному в [1-3, 6, 7].

Развитие систем обработки спутниковых данных за рубежом описано в работах [10,11]. Работа [10] посвящена организации распределенной обработки данных средствами Grid-технологии. Брокер ресурсов, получив информацию о наличии свободных ресурсов, направляет на них как обрабатываемые данные, так и сами процедуры и сценарии обработки. Особенность [11]– использование распределенных вычислений для обработки данных только одного типа (т.е. управление обработкой связано с распределением определенного количества снимков между ресурсами, выпорлняющими однотипную обработку).

Анализ способов управления обработкой показывает, что в случае наличия требуемого количества ресурсов на ВУ для данных определенного типа, диспетчирующее устройство повторно формирует сценарий обработки, который вновь передается на него для интерпретации. Это приводит к повторной активизации обрабатывающих процедур (загрузка в ОП, запуск на выполнение и т.д.). Исключить указанные недостатки позволяет подход, предусматривающий объединение однотипных данных в группы, называемые партиями. Партия данных – это совокупность данных одного типа, обрабатываемых в вычислительной системе без ее переналадки на обработку данных другого типа. Т.о. формирование партий позволяет исключить повторяющееся построение сценариев обработки для однотипных данных и передачу их на ВУ.

В силу того, что методы стандартной тематической обработки спутниковых данных предполагают ее реализацию в виде последовательности этапов (стадий), поэтому для снижения временных затрат на выполнение она должна быть конвейеризирована. В том случае, если в конвейерной системе выполняются операции с наборами спутниковых данных разных типов, тогда возникает задача управления процессом обработки. Задача управления процессом обработки является комплексной и предполагает решение двух подзадач: определения составов партий данных и построения расписаний их обработки.

Анализ современных методов решения задач определения составов партий и расписаний их обработки представлен в работе [12]. Для решения задач определения эффективных составов партий используются подходы на основе частично целочисленного линейного программирования (ЧЦЛП), эвристической процедуры и алгоритмы локальной оптимизации для задач в упрощенной постановке (один обрабатывающий прибор, директивные сроки обработки партий). Однако упомянутые методы обладают рядом недостатков: 1) для аппарата ЧЦЛП – при большой размерности задачи получение решения является затруднительным, решения по составам партий формируются без учета расписаний обработки; 2) для методов формирования партий с учетом директивных сроков обработки на одном приборе – решения по составам партий не являются эффективными, а только удовлетворяют директивным срокам; 3) для эвристических процедур – не позволяют получить эффективные решения, приближающиеся к оптимальным. В силу сказанного разработка методов определения эффективных составов партий и расписаний их обработки является актуальной задачей.

Развитие современных методов построения расписаний обработки партий представлено в работах [2-11]. В [2] выполнена классификация задач управления обработкой партий (построения расписаний обработки партий). Задачи различаются по виду процесса обработки (непрерывный либо дискретный), способам представления времени моделирования (непрерывное либо дискретное), способам формирования партий и т.д. Управление обработкой партий предполагает построение расписаний для фиксированного их состава, определение количества и размера партий до реализации процедуры построения расписаний (алгоритм определения составов партий никак не связан с характеристиками оборудования и процессом обработки, составы партий определяются без связи с построением расписаний), определение размеров партий совместно с решением задачи построения расписаний их обработки. Методы, рассмотренные в [2], предполагают формирование партий и распределение их по обрабатывающим приборам при дискретном времени моделирования для непрерывного производства. При этом использован аппарат частично целочисленного линейного программирования (ЧЦЛП). Большая размерность модели (наличие более тридцати ограничений), использование методов ЧЦЛП, задание ограниченного количества обрабатывающих приборов (ограниченного количества ресурсов) делают затруднительным определение решения для задач большой размерности за ограниченное время. Методы управления обработкой партий, рассматриваемые в [3], предполагают наличие фиксированных партий данных и их обработку на ограниченном количестве приборов (задача определения оптимальных составов партий в этой работе не рассматривается). В работах [4,5] рассматривается задача управления обработкой партий в непрерывном (химическом) производстве. Под партиями в этом случае подразумевается объемы материалов, участвующих в процессе производства (объемы партий определяются способом производства требуемого вида продукта, количеством конечного продукта, определяемым спросом). Таким образом, в [4,5] реализуется распределение обработки партий материалов различных видов (размер партий не оптимизируется), обеспечивающих выпуск продуктов разных типов, по параллельно действующим машинам. При этом рассматриваются задачи ограниченной размерности (2 продукта, 2-3 прибора в обрабатывающей конвейерной системе). В работе [6] рассматривается решение задачи среднесрочного планирования выпуска продукции при ограниченном количестве ресурсов и последующего составления расписаний обработки сформированных при планировании партий. При этом под партией подразумевается совокупность изделий одного типа, выпуск которых закрепляется за определенным предприятием или производственным участком. Размеры партий определяются в соответствии с заказами на производство и директивными сроками их выпуска. Модель оптимизации составов партий учитывает только стоимостные параметры выпуска продукции, но не учитывает временные характеристики и особенности технологических процессов. На основе полученного решения по распределению заказов по производственным участкам решается задача выделения для них ресурсов с целью обработки. Совместно задача определения составов партий и управления выпуском (формирования расписаний обработки партий) в работе не решается. В [6] использованы модель большой размерности и аппарат ЧЦЛП, что ограничивает размерность решаемой задачи. В работе [7] решается задача определения количества и составов партий единичных (разнотипных) требований, обрабатываемых на одном приборе, с заданными директивными сроками обработки и стоимостью доставки партий. Составы партий разнотипных требований формируются с учетом директивных сроков. Задача предполагает наличие одного обрабатывающего прибора, для которого отсутствуют простои при обработке партий. Формирование партий предполагает и одновременное автоматическое построение расписания их обработки. В результате решается задача определения составов партий на основе заданных директивных сроков окончания обработки входящих в них требований. Аналогичная задача формирования партий требований в соответствии с их директивными сроками завершения, обрабатываемых на параллельных машинах, рассматривается в [8]. Объединение требований в партии реализуется с использованием эвристической процедуры в соответствии со значениями параметров начала обработки и директивными сроками окончания обработки. Т.к. обработка двух партий на одной машине (приборе) не может пересекаться, тогда сроки начала и окончания обработки партий используются для их распределения по параллельным машинам (расписание вытекает из сформированных составов партий). В работе [9] решается задача планирования производства полупроводников, предполагающая совместное формирование составов партий обрабатываемых пластин разных типов и расписаний обработки партий в конвейерной системе с параллельно действующими машинами. При определении локально оптимальных решений по составам партий и расписаниям их обработки используется один обобщенный критерий. Особенностью аппарата формирования составов партий является использование метода отжига. Реализуется поиск в окрестностях текущего локально эффективного решения. Для формирования новых решений, входящих в окрестности, введены эвристические правила, оперирующие со случайно выбираемыми партиями, изменяющие как составы партий, так и расписания их обработки (изменение позиции партии, изменение обрабатывающего прибора для выполнения операции, создание новых партий). На основе сформированного совместного решения по составам партий и расписаниям их обработки выполняется оценка эффективности расписания с использованием дизъюнктивного графа. Предложенный в [9] подход позволяет реализовывать стохастический поиск локально оптимальных решений. Планированию производственного процесса посвящена работа [10], в которой решается задача распределения заказов на производство продукции по сменным заданиям, распределения заданий по партиям и формирование расписаний обработки партий. Для определения составов сменных заданий используется эвристическая процедура, для определения составов партий разработана имитационная процедура, позволяющая моделировать прохождение партиями конвейерной системы. Определение эффективных составов партий предполагает задание параметров имитационной модели, соответствующих размерам партий изделий каждого типа, и проведения моделирования. В соответствии с результатами моделирования выбираются значения параметров размеров партий, обеспечивающие минимальное значение критерия. После того как с использованием моделирования определены оптимальные размеры партий, выполняется формирование расписаний их обработки. В [11] реализуется решение задачи определения составов партий компонент, из которых выполняется формирование элементов (аналог формирования комплектов из обработанных в системе изделий). Для формирования партий и расписаний их обработки введена оптимизационная модель, являющаяся многопараметрической и многоиндексной. Определение решений по количеству и составам партий осуществляется путем полного перебора возможных значений этих параметров. Расписание для полученного решения формируется посредством использования эвристической процедуры. При большой размерности задачи (значительное число типов компонент и количество компонент каждого типа) прямой перебор при формировании партий не обеспечивает решение поставленной задачи за ограниченное время. В итоге решение задачи определения составов партий и построения расписаний их обработки реализуется путем привлечения: 1) аппарата ЧЦЛП (однако при большой размерности задачи получение решения за ограниченное время является затруднительным, решения по составам партий формируются без учета расписаний их обработки); 2) методов формирования партий с учетом директивных сроков окончания обработки входящих в них требований; 3) эвристических процедур и правил (однако, применение правил не позволяет получить решения, приближающиеся к оптимальным). Т.е. эффективные составы партий определяются с привлечением аппарата ЧЦЛП, что делает затруднительным решение задач большой размерности. Либо составы партий определяются, исходя из директивных сроков окончания обработки данных, что не предполагает оптимизацию. Либо составы партий определяются на основе эвристических правил, что также затрудняет получение эффективных решений. В соответствии с этим разработка моделей и методов определения локально оптимальных составов партий и расписаний их обработки является актуальной задачей.

Целью работы является совершенствование методов локальной оптимизации решений по составам партий данных, обрабатываемых в конвейерной системе, и решений по порядкам обработки этих партий на сегментах в конвейерной системе (расписанию обработки партий). Достижение цели обеспечивается декомпозицией обобщенной функции (цели) системы на совокупность иерархически упорядоченных подфункций (подцелей), каждая из которых реализуется на определенном уровне иерархии системы. В результате выполненной декомпозиции обобщенной функции системы в рассмотрение введена двухуровневая модель иерархической игры определения локально оптимальных решений по составам партий и расписаниям их обработки. На основании предложенного иерархического подхода достижение сформулированной цели обеспечивается решением следующих задач: а) обоснования математической модели вычислительного процесса обработки партий данных в конвейерной системе; б) обоснования вида критериев оптимизации в модели иерархической игры, используемой для определения эффективных решений по составам партий данных и расписаниям их обработки; в) обоснование метода локальной оптимизации решений по составам партий данных; г) обоснование метода локальной оптимизации решений по расписаниям обработки партий.

В рассматриваемой работе реализуется решение задачи обоснования математической модели вычислительного процесса обработки партий данных в конвейерной системе при формировании комплектов из результатов обработки и задании директивных сроков формирования комплектов каждого типа, а также задачи обоснования вида критериев оптимизации в модели иерархической игры, используемой для определения эффективных решений по составам партий данных и расписаниям их обработки.

Задача, близкая к рассматриваемой в данной работе, решалась в [12]. Особенностью постановки задачи, решаемой в [12], является наличие временных ограничений на длительности интервалов времени обработки партий данных, т.е. задание временных интервалов, в течение которых выполняется обработка партий. Из данных, обработанных в течение этих интервалов времени, формируются комплекты установленного состава. Для решения этой задачи в [12] выполнено обоснования метода формирования комплектов из результатов обработки данных, полученных по истечении каждого установленного интервала времени функционирования системы. При этом оценка эффективности решений по составам партий выполняется с точки зрения количества комплектов, сформированных из результатов обработки данных, полученных по истечении заданных временных интервалов.

Декомпозиция обобщенной функции системы управления вычислительным процессом обработки партий при задании ограничений на интервалы времени реализации операций с данными и условии формирования комплектов, выполненная в [12], позволила определить три уровня иерархии принятия решений в системе. На первом уровне осуществляется формирование решений по составам партий данных, на втором –формирование решений по расписаниям обработки партий данных в конвейерных системах. Для каждого уровня определены формы критериев эффективности принятия решений и методы формирования соответствующих решений.

Рассматриваемая в предлагаемой работе задача также предполагает определение решений по составам партий данных и порядкам их обработки в конвейерной системе (расписаниям обработки партий в конвейерной системе) при условии формирования комплектов из результатов. Однако ее особенностью является задание директивных сроков формирования комплектов каждого типа. Т.е. постановка задачи предполагает задание количества комплектов каждого типа, которые должны быть сформированы из результатов обработки данных, и задание директивных сроков формирования каждого комплекта определенного типа (моментов времени, к наступлению которых каждый комплект определенного типа должен быть сформирован). В соответствии с этим определение количества сформированных комплектов реализуется не с учетом результатов обработки, полученным к моментам времени окончания заданных интервалов времени (как в работе [12]), а с учетом результатов, формируемых по мере обработки данных (в течение всего времени обработки партий). При этом моменты времени окончания формирования каждого комплекта определенного типа должны соответствовать директивным сроком, для них заданных.

Конвейеризация программ обработки данных предполагает, что за каждым сегментов конвейера закреплены для выполнения определенные их части [1]. Введем в рассмотрение следующие обозначения: *i –* идентификатор типа данных, обрабатываемых в системе, *n –* количество типов обрабатываемых данных (), – количество данных *i-*го типа, которые должны быть обработаны. Данные *i-*го типа обрабатываются соответствующей им программой, тогда *i* – идентификатор программы, выполняемой в составе конвейера, обрабатывающей данные *i-*го типа. Однократное выполнение конвейеризированной программы *i-*го типа обеспечивает обработку одного элемента данных *i-*го типа. Если значение (), тогда обрабатывающая программа должна быть выполнена в конвейерной системе раз. Производительность всех сегментов конвейера является одинаковой, объемы вычислений, выполняемые в соответствии с программным кодом, реализующим обработку данных *i-*ых типов () на каждом сегменте, различны. Тогда функционирование каждого сегмента конвейера при обработке данных характеризуется длительностями выполнения им операций с данными каждого *i-*го типа. Т.к. являются различными длительности выполнения программ на соответствующих сегментах, тогда может быть сформировано расписание обработки данных, представляющее собой порядок запуска программ на выполнение. В результате управление вычислительным процессом в конвейерных системах предполагает определение порядка запуска программ обработки данных на выполнение.

Постановка задачи предполагает задание значений (). Однотипные данные могут быть объединены для обработки в партии. Партия–это совокупность данных одного типа, которые обрабатываются без переналадки (перенастройки) конвейера. Переналадка (перенастройка) сегмента конвейера представляет собой завершение активного состояния программы, выполнявшей обработку данных, и активизацию программы, которая будет реализовывать обработку данных в соответствии с сформированным расписанием. Характеристиками партии являются: тип *i* и количество данных в ней. Партия является фиксированной, если в нее входят все данные *i-*го типа. Если партия содержит не все данных, в системе обрабатывается ни одна, а несколько партий данных этого типа.

Особенностью постановки задачи является требование формирования комплектов разных типов из результатов обработки (составы комплектов разных типов являются заданными). Комплект данных (результатов обработки) – это совокупность результатов обработки разных типов, входящих в него в заданном количестве. Т.е. составы комплектов являются заданными и не могут быть изменены. При формировании комплектов предполагается, что данные всех *n* типов в определенном количестве входят в комплекты каждого типа. Комплекты заданных составов формируются из результатов по мере обработки данных в конвейерной системе (данные, входящие в обрабатываемую партию, после обработки снимаются с выхода конвейера и могут быть использованы при формировании комплекта соответствующего типа). Тогда могут быть зафиксированы моменты времени окончания формирования каждого комплекта соответствующего типа.

В соответствии с выполненными рассуждениями управление вычислительным процессом обработки партий данных в конвейерной системе при условии формирования комплектов заданных составов и заданных директивных сроках их формирования предполагает: 1)определение составов партий данных, обрабатываемых в системе; 2) определение порядка обработки партий данных; 3) распределение обработанных в составе партий данных по комплектам заданного состава. Для решения поставленной задачи определения составов партий и порядков их обработки на сегментах конвейера применены методы, реализующие локальную оптимизацию решений (это связано с доказанным свойством NP-полноты рассматриваемой задачи). Поэтому эффективными решениями по составам партий и порядкам их обработки будут являться те решения, которые при реализации локального поиска позволяют получить минимальные значения введенных критериев оптимизации (локально эффективные решения).

Рассматриваемая задача является задачей с полной информацией, все параметры, характеризующие обрабатываемые данные (типы данных, количество данных, длительности обработки данных различных типов, составы комплектов различных типов и т.д.) и функционирующую систему (количество сегментов, дисциплина обработки партий, длительности интервалов времени функционирования системы и т.д.) являются заданными. Т.к. черезi обозначен идентификатор типа данных, обрабатываемых в системе, тогда через обозначим момент времени поступления в систему данных каждого i-го типа (). Для всех программ моменты времени их загрузки в систему и моменты поступления данных на обработку одинаковы, при этом . Обозначим через l индекс сегмента конвейера, осуществляющего выполнение l-й части программы, при этом . Каждым сегментом конвейера выполняются вычисления, соответствующие назначенной для него части программы. Дисциплина обслуживания выполняемых в системе программ предполагает прохождение данными, которые они обрабатывают, всех сегментов конвейера, при этом прерывание обработки данных является недопустимым. Выполнение на каждом l-м сегменте назначенной ему части i-го программы характеризуется параметром длительности обработки данных, однозначно соответствующей объему вычислений.

Классификации решаемой задачи построения расписаний выполнена по аналогии с [12]. В соответствии с [13] задача составления расписаний может быть представлена в следующем виде: , где параметр соответствует классу (типу) обрабатывающей системы, – типу решаемой задачи и характеризует ее особенности, – виду критерия оптимизации при определении эффективного решения. Рассматривается конвейерная система, которая в [3] и [13] определена как *FlowShop*. Тогда в качестве указывается *F*. Решаемая в системе задача обработки данных предполагает объединение обрабатываемых данных в партии. В этом случае в соответствии с [3] и [13] в качестве характеристики решаемой задачи указывается *batch*. Критерий, в соответствии с которым определяется эффективное расписание, учитывает особенности метода определения порядка обработки партий, реализующего жадный подход. Метод предполагает, что в последовательности обработки партий, порядок выполнения операций с которыми оптимизирован на предыдущих шагах алгоритма, добавляется новая партия данных и определяется ее эффективное местоположение в этих последовательностях. Критерий эффективности решения по порядку обработки партий данных сформирован с точки зрения условия реализации внутренней цели функционирования системы, которая обуславливает необходимость эффективного использования оборудования системы (эффективное использование оборудования системы связано с минимизацией простоев сегментов конвейера при реализации обработки партий). Для классификации задачи в рассмотрение введено обозначение простоя (ожидания) отдельного *l*-го сегмента конвейера при обработке им партий в виде . Критерий, используемый при определении локально эффективного расписания обработки партий, учитывает суммарные простои всех сегментов конвейера при реализации операций. Для классификации задачи критерий обозначен в виде . Тогда рассматриваемая задача с точки зрения [3], [13] может быть охарактеризована как: .

Особенностью решаемой задачи является необходимость формирования комплектов из результатов обработки данных и задание директивных сроков окончания формирования каждого комплекта определенного типа. Комплект результатов обработки данных представляет собой совокупность данных, при этом количество данных каждого типа, входящих в комплект одного типа является заданным. Через *g* обозначим идентификатор типа комплекта, который формируется из результатов обработки данных в конвейерной системе, а через – общее количество типов формируемых комплектов. Через обозначим количество данных *i*-го типа, которые должны входить в один комплект *g*-го типа. В рассмотрение введена матрица (*W*)*,* значения элементов которой равны числу данных *i*-го типа, которые входят в один комплект *g*-го типа. Через обозначим количество комплектов *g-*го типа, которые должны быть сформированы из обрабатываемых данных *i-*ых типов () в соответствии с заданием. Тогда количество данных каждого *i-*го типа (), которые будут обработаны в системе, определяется следующим образом: . В итоге все поступившие на вход системы для обработки данные будут использованы при формировании комплектов разных типов. Также через обозначим индекс (номер) сформированного комплекта *g-*го типа, тогда . Т.к. решаемая задача предполагает задание для каждого комплекта директивных сроков его формирования, тогда в рассмотрение введена матрица , элемент которой представляет собой задаваемый момент времени окончания формирования -го комплекта *g-*го типа (,). При этом для каждого g-го типа элементы матрицы упорядочены в ее *g-*ой строке по возрастанию значений моментов времени окончания формирования комплектов: , если . Тогда обработка данных в конвейерной системе направлена на осуществление операций на всех сегментах конвейера с данными разных типов таким образом, чтобы формирование комплектов из результатов обработки выполнялось с учетом (в соответствии) с задаваемыми директивными сроками.

Определение эффективных решений по составам партий данных *i-*ых типов () вместо фиксированных партий позволяет получить лучший результат с точки зрения формирования в установленные сроки комплектов из результатов обработки. При этом достигается внешняя цель функционирования системы – обеспечение в заданные сроки формирования комплектов установленных типов из результатов обработки данных в конвейерной системе. При этом должно быть определено как количество партий данных, так и их составы.

Входными данными для системы построения расписаний обработки партий при формировании комплектов являются: типы *i* () данных, обрабатываемых в системе, количество данных каждого *i*-го типа; матрица *(W)*,элемент которой равен количеству данных *i*-го типа, которые должны входить в один формируемый комплект *g*-ого типа ( ), матрица директивных сроков формирования комплектов, элемент которой соответствует задаваемому моменту времени окончания формирования -го комплекта *g-*го типа (,). Выходными решениями, обеспечивающими формирование комплектов в установленные сроки, являются: количество и составы партий данных *i*-ых типов (); расписания обработки партий данных в конвейерной системе.

В итоге обобщенная цель функционирования системы представлена как совокупность иерархически упорядоченных целей подсистем (функций, решаемых на иерархически упорядоченных уровнях системы построения расписаний обработки партий). Формирование решений на уровнях системы построения расписаний обработки партий осуществляется следующим образом: первый уровень – решения по количеству и составам партий, второй уровень – решения по порядку обработки партий сформированного состава на сегментах конвейера. Выполненная декомпозиция обобщенной функции системы на совокупность иерархически упорядоченных подфункций позволяет рассмотреть задачу как задачу теории иерархических игр [14,15].

С целью формирования модели иерархической игры для построения комплексных расписаний обработки партий в рассмотрение введены следующие обозначения: – количество партий данных *i*-го типа (), формируемых на первом уровне принятия решений, при элементы образуют вектор *(М)*; *А–* матрица, элемент которой – это количество данных *i*-го типа в *h-*ой партии (), размерность матрицы *А –* , где . Решение, формируемое на первом уровне системы (количество и составы партии) имеет вид: *[М, А]*.

Расписание обработки партий данных *i-*ых типов () обозначено как , оно представляет собой совокупность (множество) последовательностей запуска партий на обработку на каждом *l-*ом сегменте конвейера (). Расписание имеет вид: . Расписание обработки партий формируется в предположении, что порядок обработки партий является одинаковым на всех *L* сегментах конвейера. Для формализации вида последовательностей расписания в рассмотрении введена матрица порядка обработки партий в системе; элемент , если партия данных *i*-го типа занимает в последовательности  *j*-ю позицию, в случае, если партий данных *i-*го типа не занимает в последовательности  *j*-ю позицию, размерность матрицы , где – количество партий в последовательностях расписания . Порядок обработки партий группы на всех сегментах одинаков, поэтому достаточно определения одной матрицы порядка . В рассмотрение введена матрица – матрица количества данных *i*-ых типов в партиях, занимающих в последовательностях *j*-е позиции (элемент равен количеству данных *i*-го типа в партии, занимающей *j*-ю позицию в , размерность матрицы). Решение, формируемое на нижнем уровне иерархии, имеет вид: .

В соответствии с функциями уровней системы между ними выполняется обмен информацией следующего вида: 1) на вход первого уровня подаются: количество *n* типов обрабатываемых данных; количество () данных каждого типа, обрабатываемых в системе (определяемое на основе заданных значений количества комплектов каждого *g*-го типа ()); матрица *(W)* составов комплектов; матрица директивных сроков окончания формирования комплектов *g-*ых типов (); выхода уровня передаются составы партий данных *i*-ых типов () – решение *[М, А]*; 2) на вход второго уровня – решение *[М, А]*; с выхода уровня – сформированное для решения *[М, А]* эффективное расписание обработки партий. При формировании расписаний обработки партий данных *i-*ых типов состав партий не изменяется (значения и , поступившие с верхнего уровня, изменены быть не могут). В общем виде модель иерархической игры в соответствии с введенными обозначениями имеет следующий вид [14,15]:

1) первые уровень:

, (1)

2) второй уровень:

(2)

С точки зрения аппарата теории иерархических игр каждым игроком принимается решение, соответствующее функции (подцели) отдельного уровня. Игрок на верхнем уровне принимает решение по составам партий данных, игрок на втором уровне принимает решение по расписанию обработки партий данных в конвейерной системе.

Комплекты формируются только из результатов обработки данных, которые входят в партии. Полное формирование одного комплекта *g*-го типа предполагает получение (в требуемом количестве) всех результатов обработки данных, для этого необходимых. Формирование всех результатов обработки данных (в требуемом количестве) требуется реализовать к директивному сроку формирования комплекта соответствующего типа. Поэтому наилучшим решением по составам партий будет являться то решение (с учетом построенного расписания обработки этих партий), которое обеспечивает формирование комплектов разных типов в соответствии с их директивными сроками. Для определения эффективных решений на первом уровне должно определяться соответствие сроков окончания формирования комплектов (для решений по составам партий и расписанию их обработки) заданным директивным срокам. Данное соответствие характеризуется степенью превышения реальных сроков формирования комплектов заданным директивным срокам. Это условие соответствует внешней цели функционирования системы и интерпретируется при формировании критерия эффективности решений на верхнем уровне иерархии.

Обоснование метода определения эффективных составов партий данных

Для обоснования метода формирования локально оптимальных решений по составам партий данных *i-*ых типов () в рассмотрение введены следующие обозначения: – идентификатор типа данных, составы партий которого изменяются на текущем шаге алгоритма; – количество партий данных -го типа, составы которых определяются; *s*– индекс текущего улучшаемого решения путем поиска более эффективного решения в рамках окрестности; *g –* индекс шага алгоритма, выполняемого по отношению к шагу *s*, который соответствует новому формируемому решению, находящемуся в окрестности (*(s+g)*– номер шага алгоритма по формированию решения в окрестности текущего локально оптимального решения); – индекс партии, состав которой изменяется при реализации алгоритма на текущей итерации; *I –* множество типов данных, для которых выполняется формирование составов партий (первоначально ); – множествотипов данных, используемое при формировании локально оптимального решения по составам партий; *A’*– матрица (аналог матрицы составов партий *А*), используемая при определении наилучшего решения в окрестности текущего локально оптимального решения (матрица *А’* используется при формировании промежуточных решений); и ()– матрицы, предназначенные для хранения (буферизации) составов партий данных *i*-ых типов, сформированных на различных итерациях алгоритма (в матрицах () храняться решения по составам партий данных *i-*ых типов, входящие в окрестность с меньшей метрикой, в матрицах () храняться решения по составам партий данных *i-*ых типов, входящие в окрестность , сформированные на основе решений из окрестности , т.е. на основе решений из матриц ()); – количество решений по составам партий данных *i*-го типа в матрице размерностью ; – количество решений по составам партий данных *i*-го типа, в матрице размерностью ; – индекс текущей строки (решения по составам партий *i-*го типа) в матрице (); – индекс текущей строки (решения по составам партий *i*-го типа) в матрице (); – параметр, предназначенный для хранения номера строки в – номера решения, гарантирующего максимальное по модулю значение левого дискретного градиента критерия [13]; *G–* максимальное по модулю значение левого дискретного градиента , достигаемое в окрестностях различного вида текущего локально оптимального решения (максимального по модулю значение ).

Для текущего локально оптимального решения по составам партий данных *i-*ых типов рассматриваются два вида окрестностей. Построение решений, включаемых в окрестность первого вида, связано с: а) изменением составов партий данных каждого *i-*го типа в заданном количестве () при неизменных составах партий данных других типов; б) увеличением количества партий данных *i-*го типа в случае, если построение решений путем изменения составов партий данных этого типа в количестве является невозможным (выполнены условия окончания формирования составов партий в количестве ). Тогда реализуется изменение количества партий данных *i-*го типа и задание начального решения для этого количества партий. Построение решений, включаемых в окрестность второго вида, предполагает совместное использование сформированных решений по составам партий данных всех типов при заданном количестве партий (реализуется совместное рассмотрение решений по составам партий данных разных типов). Для обоснования алгоритма определения эффективных решений сформулирован способ изменения составов партий данных *i’-*го типа.

Для обоснования способа формирования решений по составам партий данных введены следующие условия и рассуждения (для *i-*го типа данных):

1) количество данных -готипа в партиях должно быть не менее 2 (,); если при формировании начального решения по составам партий данных -готипа для *h*=1 (первая партия) получено , тогда исследование решений по составам партий данных этого *-*го типа не выполняется;

2) способ формирования начального решения для количества партий данных *-*го типа предполагает, что (), а элемент определяется как ; в дальнейшем при формировании составов партий данные извлекаются из партии с *h*=1 и распределяются по остальным партиям ();

3) значения параметров , задаваемые первоначально для данных всех *i-*ых типов (), равны 2 ();

4) модификация количества партий -го типа данных предполагает, что параметр увеличивается (при определении составов партий) до тех пор, пока в начальном решении для выполняется условие ; при условии (для начального решения при значении ) формирование составов партий -го типа прекращается;

5) формирование решений по составам партий предполагает увеличение количества данных в партии с индексом (при неизменном составе *h*-ых партий ) и уменьшение количества данных в партии с *h=1* (); при выполнении условия модификация составов партий продолжается; при выполнении полученное решение по составам партий в количестве не рассматривается, т.к. является аналогичным полученному на предыдущих шагах алгоритма.

Способ модификации составов партий *i’-*го типа предусматривает, что на основе решений, находящихся в окрестности первого вида текущего локально оптимального, формируются решения, находящиеся в окрестность с большей метрикой. Метрика *k* окрестности , содержащей решения по составам партий *i’*-го типа, определяется выражением , где – элемент *i’*-ой строки матрицы , соответствующей текущему локально оптимальному решению по составам партий данных, – элемент *i’*-ой строки матрицы , соответствующей решению по составам партий в окрестности .

В матрице выполняется хранение решений, находящихся в окрестности первого вида , на основе которых реализуется формирование решений, входящих в окрестность (в матрице хранятся решения, используемые при формировании решений в окрестности ). В матрице реализуется хранение сформированных решений, находящихся в окрестности . Тогда на основе одного решения по составам партий данных (-ой строки матрицы ), формируется совокупность решений, хранимых в матрице , находящихся в окрестности . Если для текущего рассматриваемого решения (-ой строки матрицы ) выполняется условие , тогда увеличение на 1 значения (при неизменном значении ), а затем увеличение на 1 значения (при неизменном значении ) обуславливает получение одинаковых решений по составам партий данных *i-*го типа. Таким образом, формулировака условия, ограничивающего количество решений в окрестности , имеет следующий вид: если при формировании нового решения по составам партий данных -го типа на основе решения, представленного строкой матрицы , для -ой строки выполняется условие , тогда не требуется выполнять увеличение значения элемента ; должен быть выполнен переход к элементу , который увеличивается в случае выполнения условия .

Описание алгоритма определяющего локальное оптимальное решение по составам партий.

Для определения локально оптимального решения по составам партий данных *i-*ых типов () сформулированы: 1) способ формирования решений по составам партий данных одного *i’-*го типа, включаемых в окрестности первого вида, на основе решений, входящих в окрестность (изменение составов партий в количестве ); 2) способ формирования решений, представляющих собой комбинацию решений по составам партий данных всех *i*-ых типов (), включаемых в окрестности второго вида; 3) обобщенный алгоритм определения локально оптимального решения по составам партий данных. Входными параметрами алгоритма формирования решений в окрестности первого вида, получаемых на основе решений из окрестности , являются: идентификатор *i’-*го типа данных, для которого определяются составы партий; количество решений в окрестности ; количество партий , составы которых изменяются; матрица , содержащая решения в окрестности . Алгоритм формирования решений по составам партий данных одного *i’*-го типа в окрестности первого вида на основе решений из окрестности имеет следующий порядок шагов:

1) инициализация значений параметров: , , , , ;

2) инициализация номера партии , состав которой изменяется, значением 2 ();

3) изменение ; в партии с индексом – увеличение количества данных на 1; построение решения по составам партий *i’*-го типа (-ой строки матрицы ): (), , ; ;

4) проверка условия ; в случае его выполнения сформированное в виде -ой строки матрицы решение не интерпретируется, тогда и (составы партий *i’*-го типа в количестве не модифицируются); при условии полученное решение интерпретируется, выполняется задание значения параметра *j=1* (*j–* значение шага изменения номера партии *h’*), реализуется переход на шаг 5;

5) проверка условия ; при реализуется проверка условия формирования решения по составам партий -го типа в виде ; если условие не выполняется, тогда реализуется переход на шаг 6; при выполнении условия существует возможность формирования решения по составам партий данных *i’-*го типа, тогда , реализуется переход на шаг 3 (переход к формированию следующего решения в окрестности ); при – переход на шаг 7;

6) в случае выполнения условия при модификации ()-ой партии будет получено решение, дублирующее полученное ранее; осуществляется модификация параметра *j=j+1*; выполняется переход на шаг 5;

7) при сформированы все решения (хранящиеся в матрице ) с использованием одного решения, хранящегося в -ой строке матрицы ; переход на шаг 8;

8) модификация значения – индекса решения, хранимого в матрице (переход к следующему решению из окрестности , на основании которого формируются новые решения): , при (рассмотрены не все решения из окрестности , хранимые в матрице ) реализуется переход на шаг 2; при условии все решения, хранящиеся в матрице , использованы при формирования решений в окрестности ; выполняется переход на шаг 9;

9) при выполняется сравнение решений, хранимых в матрице , с точки зрения дублирования ими друг друга (процедура сравнения предполагает формирование копии матрицы , в копии матрицы упорядочивание элементов каждой *-*ой строке () по убыванию их значений, поэлементное сравнение каждой *-*ой строки () с другими *-*ыми строками (), удаление из матрицы строк с индексами , которые соответствуют строкам в копии матрицы , дублирующим рассматриваемую *-*ю строку; при удалении *q2’’*-ой строки из матрицы реализуется изменение и переупорядочивание решений в матрице ); в результате повторяющиеся решения из матрицы удалены; задание значений ; при осуществляется переход на шаг 11;

10) при не были сформированы новые решения в окрестности , выполняется переход на шаг 16;

11) реализуется исследование решения по составам партий данных *i*’-го типа в окрестности с точки зрения его эффективности; для этого выполняется инициализация *i*’-ой строки матрицы *А’* решением по составам партий данных, хранимым в -ой строке матрицы : (); передача сформированного решения на второй уровень для определения расписаний обработки партий (при этом исходное решение *A(s)* не изменяется);

12) получение со второго уровня локально оптимального решения вида , использование этого решения для вычисления значения ;

13) для решения *[M(s+g),A’(s+g)]* вычисление левого (правого) дискретных градиентов (); вычисление () выполняется по формулам [14]: а) ; б) ;

14) при решение *[M(s+g),* *A’(s+g)]* не является более эффективным, чем решение ; осуществляется модификация значения ; если (рассмотрены не все решения из окрестности ), тогда выполняется модификация индекса шага алгоритма *g=g+1*, реализуется переход на шаг 11 (переход к анализу следующего решения из ); при выполняется переход на шаг 16;

15) если для решения по составам партий (-ой строки матрицы ) реализуется , тогда выполняется сравнение значений и *G* градиента целевой функции (являющегося «глобальным» для всех типов данных); если , тогда текущее решение по составу партий *i’*-го типа данных является наилучшим среди всех рассмотриваемых решений по составам партий данных *n* типов в окрестностях первого вида; при значения всех параметров () обнуляются, индекс решения (-ой строки матрицы ) сохраняется: , модификация значения *G*: ; реализуется переход к следующему решению в окрестности , для этого ; если , тогда *g=g+1*, реализуется переход на шаг 11; если , тогда выполняется переход на шаг 16;

16) если , тогда в окрестности определено решение по составам партий *i’*-го типа, лучшее, чем текущее оптимальное решение , останов алгоритма определения решений по составам партий данных *i’*-го типа в окрестности ;

Выходными параметрами алгоритма формирования составов партий каждого *i’-*го типа в окрестности являются: – количество сформированных решений по составам партий, входящих в ; значение индекса строки в матрице , соответствующей наилучшему решению по составам партий данных *i’*-го типа в окрестности ; матрица , содержащая (при ) решения, входящие в окрестность . При и в окрестности отсутствует решение, являющееся лучшим, чем текущее локально оптимальное, необходимо формирование решений в окрестности первого вида с большей метрикой. Рассмотренный способ определения решений, входящих в окрестность , последовательно применяется ко всем типам данных (). Он предполагает (для *i-*го типа, ) формирование на основе решений из матриц новых решений, сохраняемых в матрицах . Тогда в окрестность первого вида входят решения по составам партий *i-*ых типов (), сформированные путем изменения составов партий, сохраненных в матрицах .

В том случае, если в окрестности первого вида не найдено решения, лучшего, чем текущее локально оптимальное решение , тогда поиск выполняется в окрестности второго вида. Решения, входящие в окрестность второго вида, формируются путем совместного рассмотрения сформированных решений по составам партий *i*-ых типов, сохраненных в матрицах (). Формулировка способа формирования решений по составам партий данных, входящих в окрестность второго вида предваряется введением обозначений: , – множества идентификаторов типов данных, для которых при формировании решений в окрестности первого вида получено , при этом элементы этих множеств являются упорядоченными по возрастанию, – множество типов данных, для которых при формировании решений в окрестности первого вида получено , при этом элементы этого множества являются упорядоченными по убыванию.

Алгоритм формирования решений по составам партий, входящих в окрестность второго вида решения , предполагает реализацию следующих шагов:

1) формирование множеств и в виде: , , при этом типы данных и такие, что , , где *j,k*– порядковые номера идентификаторов типов данных в множествах и ; формирование множества : при , ;

2) если , тогда извлечение из  идентификатора *i’*-го типа данных следующим образом: ; модификация множества : ; задание значения параметра для рассматриваемого *i’*-го типа данных; для *i’-*го типа данных определение текущего решения по составам партий (формирование *i’*-ой строки матрицы *A’*): ();

3) проверка условия , при его выполнении осуществляется переход на шаг 2; при выполнении условия сформирована матрица *A’*, входящая в решение ;

4) передача решения на второй уровень иерархии для формирования на нем решения вида по порядку обработки партий; получение со второго уровня иерархии решения , соответствующего решению ; определение на основе значения критерия , соответствующего решению ; для решения вычисление левого (правого) дискретного градиента (); при решение не является более эффективным, чем решение , поэтому оно не интерпретируется, выполняется переход на шаг 5; при реализации выполняется сравнение значений и градиента целевой функции ; если , тогда текущее решение по составам партий данных *i*-ых типов () является наилучшим среди рассмотренных решений по составам партий данных в окрестностях второго вида, для всех *i-*ых типов данных () выполняется инициализация параметров алгоритма: (при ), модификация значения : ; реализуется переход на шаг 5;

5) извлечение из идентификатора *i’*-го типа данных, для которого изменяется решение по составам партий: ; определение позиции (номера) *j’* этого типа данных в множестве ; модификация значения параметра для рассматриваемого *i’*-го типа данных: ; если , тогда формируется *i’*-я строка матрицы *A’*: (), выполняется переход на шаг 7;

6) если , тогда ; если , тогда реализуется переход на шаг 5; при выполняется переход на шаг 8;

7) формирование множеств , в виде: , (множество включает типы данных, следующие в множестве за текущим рассматривамым *i’*-ым типом (начиная с *(j+1)*-ой позиции), если идентификатор рассматриваемого типа данных *i’* является «последним» в множестве , тогда , вид множества соответствует первоначальному); выполняется переход на шаг 2;

8) останов алгоритма.

На основе способов формирования решений по составам партий данных *i-*ых типов (), входящих в окрестности первого и второго видов, сформулирован алгоритм определения локально оптимальных решений по составам партий. Алгоритм предполагает формирование решений путем: 1) изменения составов партий данных каждого типа при неизменных составах партий других типов; увеличения количества партий данных *i-*ых типов и задания начальных составов этих партий; 2) совместного использования измененных составов партий данных всех *n* типов, определение окрестности решений второго вида при неизменных значениях параметров .

Перед реализацией алгоритма определения локально оптимальных решений по соствам партий данных выполняется инициализация множества типов данных, для которых формируются составы партий, в виде: , инициализация индекса шага алгоритма *s* и значения метрики окрестности *k* значением 0.

Алгоритм определения локально оптимального решения по составам партий данных всех *i-*ых типов () содержит следующий порядок шагов:

1) задание для каждого *i-*го типа данных значений параметров () равными 1 (– количество решений по составам партий данных *i*-ых типов, сформированных на текущей итерации, хранимых в матрице (буферизируемых в матрице ));

2) формирование начального решения по составам партий данных *i-*ых типов (): а) (); б) инициализация *h*-ых элементов *i*-ых строк матрицы *А*: (), ; в) проверка условия корректности решений по составам партий данных -ых типов (): и ; г) если для какого-либо *i*-го типа данных () в начальном решении это условие не выполняется, тогда для данных этого типа формируется фиксированная партия: , , ; модификация множества *I* типов данных: ; параметр *G* инициализируется значением 0; инициализация множества :; результатом является сформированное начальное решение по составам партий данных ;

3) передача решения на второй уровень для формирования расписания обработки партий; получение со второго уровня расписания для текущего решения по составам партий в виде ; определение значения для решения ;

4) определение -го типа данных, составы партий которого изменяются: ; ;

5) для определения модифицированных составов партий данных *i’-*го типа (формирования решений в окрестности текущего локально оптимального решения ) реализуется инициализация: параметра , элементов матрицы для *i’-*го типа данных ();

6) определение решений по составам партий данных *i’*-го типа, входящих в окрестность первого вида, формируемых на основе решений, находящихся в матрице (для этого вызывается алгоритм формирования составов партий данных *i’*-го типа, рассмотренный выше, использующий решения, хранящиеся в матрице , для формирования новых решений в окрестности первого вида); при выполнении для рассматриваемого *i’-*го типа данных условия реализуется переход на шаг 14;

7) если для данных *i’*-го типа после формирования решений в окрестности первого вида выполняется условие (в окрестности текущего локально оптимального решения не получены решения, для которых выполняется условие корректных составов партий), тогда реализуется модификация количества партий: ; выполняется формирование начального решения по составам партий в количестве : осуществляется инициализация значений ; , , , элементов матрицы : (), ;

8) проверка корректности сформированного начального решения по составам партий данных в количестве : ; (); при его выполнении реализуется переход на шаг 9; если условие корректных составов партий для начального решения не выполняется, тогда формирование новых решений по составам партий данных *i’*-го типа является невозможным, текущая *i’*-ая строка матрицы *A* соответствует эффективному решению по составам партий данных этого типа; удаление *i’*-го типа данных из множества *I*: (формирование решений по составам партий данных для *i’-*го типа данных прекращается), переход на шаг 4;

9) анализ эффективности решения по составам партий данных в количестве (решение в окрестности первого вида, ), для этого выполняется инициализация *i*’-ой строки матрицы *А’* решением по составам партий данных, хранимым в строке матрицы : ( при ); передача сформированного решения для определения расписания обработки;

10) получение для рассматриваемого решения со второго уровня расписания обработки партий в виде , его использование для вычисления значения критерия ;

11) для решения *[M(s+g),A’(s+g)]* вычисление левого (либо правого ) дискретного градиентов целевой функции [14];

12) при условиию сформированное решение *[M(s+g),* *A’(s+g)]* не является более эффективным, чем решение (для количества партий ), тогда реализуется переход на шаг 14 (параметр не инициализируется);

13) если для сформированного начального решения по составам партий (строки матрицы ) выполняется условие , тогда реализуется сравнение значений и *G*– градиента целевой функции ; если , тогда текущее решение по составам партий данных *i’*-го типа является наилучшим среди рассмотренных решений по составам партий данных в окрестности первого вида; если, тогда значения всех параметров () обнуляются, индекс решения по составам партий данных *i’*-го типа (-ой строки матрицы ) сохраняется: ; изменение значения *G*: , переход на шаг 14;

14) осуществляется проверка условия ; в случае его выполнения реализуется переход на шаг 4; при (для каждого *i*-го типа данных () сформированы новые решения по составам партий) осуществляется переход на шаг 15;

15) проверка условия , при его выполнении отсутствуют *i-*е типы данных (), для которых сформированные решения удовлетворяют условию корректного состава партий (отсутствует возможность анализа новых решений), выполняется переход на шаг 24; при для *i-*ых типов данных () реализуется проверка условия :; при выполнении для *i’-*го типа данных () условия решение по составам его партий (при неизменных составах партий других *i*-ых типов (, )) обеспечивает максимальное уменьшение значения целевой функции в рамках окрестности первого вида; реализуется переход на шаг 16; в случае, если для всех *i-*ых типов данных () выполняется условие , тогда в рамках окрестности первого вида не определено решение, улучшающее решение ; переход на шаг 17;

16) формирование локально оптимального решения с учетом составов партий данных *i’-*го типа, для которого : инициализация *i’*-ой строки матрицы *A* текущего решения значениями элементов -ой строки матрицы : (); в результате получено новое локально оптимальное решение ; инициализация ();; переход на шаг 4;

17) формирование решений по составам партий данных *i*-ых типов, включаемых в окрестность второго вида, путем комбинирования решений по составам партий данных *i-*ых типов, находящихся в матрицах ; для этого вызывается алгоритм формирования составов партий данных *i*-ых типов, рассмотренный выше, использующий решения, хранящиеся в матрицах , для формирования новых решений в окрестности второго вида и определяющий лучшее из них в этой окрестности;

18) для *i-*ых типов данных () проверка условий ; при выполнении этих условий в окрестности второго вида не сформировано решение лучшее, чем текущее локально оптимальное решение ; тогда реализуется переход на шаг 20; если для *i-*ых типов () условия не выполняются, тогда в окрестности второго вида сформировано решение, являющееся лучшим, чем текущее локально оптимальное решение ; реализуется переход на шаг 19;

19) выполняется формирование нового локально оптимального решения с учетом составов партий данных *i*-ых типов, для которых ; для каждого *i* идентификатора типа данных, для которого выполняется условие реализуется: инициализация *i*-ой строки матрицы *A* решения  значениями элементов -ой строки  матрицы :  при ; результат– новое локально оптимальное решение ; инициализация  (), ; переход на шаг 4;

20) инициализация множества типов данных :;

21) идентификация *i’-*го типа данных, решения по составам партий данных которого буферизируются в матрице : , ;

22) инициализация строк матрицы решениями по составам партий данных *i’-*го типа, находящимися в матрице , в количестве : (, , ,), присваивание ; в результате сформирована матрица , содержащая решения, на основе которых будут формироваться решения в окрестности первого вида с большей метрикой;

23) проверка условия , если условие выполняется, тогда реализуется переход на шаг 21; в случае выполнения условия промежуточные решения по составам партий данных всех *i-*ых типов () буферизированы в соответствующих матрицах ; осуществляется присваивание , *k=k+1* , реализуется переход на шаг 4;

24) окончание алгоритма.

Вычислительная сложность сформулированного алгоритма определения локально оптимальных решений по составам партий составляет . Точность приближения получаемых решений к глобально оптимальному =0.001.

Метод определения составов партий данных *i-*ых типов () взаимодействует с методом формирования расписаний обработки партий, рассмотренным в [13].

Основной компонентой иерархической системы построения комплексных расписаний обработки партий является подсистема формирования составов партий, функционирующая на верхнем уровне. Для анализа эффективности метода определения локально оптимальных решений по составам партий в рассмотрение введены следующие обозначения параметров задачи: – отношение максимальной длительности обработки данных *i-*ых типов () на всех *l-*ых сегментах конвейера () к минимальной длительности обработки данных этих типов; – отношение максимальной длительности переналадки сегментов конвейера () к минимальной длительности переналадки этих сегментов. При исследовании параметр *n* принимал значения 5, 10 (при увеличении значений этого параметра тенденция изменения эффективности метода определения составов партий сохраняется), параметр () принимал значения 12, 16, 24. Значения параметра *L* задавались равными 5, 10, значения отношения задавались равными: 1, 2, 4, 8, 16. Аналогичным образом значения отношения задавались равными 1, 2, 4, 8, 16.

При исследовании эффективности метода построения решений по составам партий данных для определенного набора исходных входных характеристик определялись решения по обработке:

1) фиксированных партий (набор данных определенного *i-*го типа в количестве интерпретировался как одна фиксированная партия);

2) не фиксированных партий (формировались эффективные решения по составам партий данных и расписания их обработки).

В качестве параметра эффективности полученных решений определялось отношение разности значений моментов времени окончания обработки фиксированных партий и партий, сформированных в соответствии с предложенным методом, к моменту времени окончания обработки фиксированных партий (соответствующие зависимости на Рис. 1-3).

На основе полученных оценок эффективности определения решений по составам партий данных определены следующие особенности процесса оптимизации:

1) при одинаковых значениях *n* и увеличение количества приборов *L* обуславливает снижение эффективности процесса определения составов партий данных;

2) увеличение значения при прочих неизменных значениях входных параметров обуславливает увеличение эффективности метода формирования решений по составам партий;

3) увеличение значений отношения обуславливает увеличение эффективности формирования решений по составам партий данных, т.е. увеличение неоднородности в длительностях переналадок приборов обуславливает рост эффективности метода формирования решений по составам партий данных;

4) увеличение значений отношения при зафиксированных значениях отношения обуславливает увеличение эффективности решений по составам партий данных.

Также реализовано сравнение результатов выполненных исследований по определению эффективности разработанного метода формирования составов партий с результатами других методов определения решения задачи (в частности, с генетическими алгоритмами (ГА)). Для этого применен алгоритм, предложенный в [13]. Аппарат ГА выбран в качестве способа получения решений рассматриваемой задачи, т.к. он является достаточно проработанным к настоящему времени и позволяющим получать достаточно эффективные решения.

Обоснование способов определения моментов времени окончания формирования комплектов из результатов обработки данных в партиях в соответствии с расписанием

Распределение результатов обработки данных по комплектам реализуется с учетом порядка обработки партий в расписаниях, сформированных для определенного решения по их составам. Т.е. комплекты формируются по мере готовности результатов, полученных при обработке партий данных. При реализации распределения результатов обработки по комплектам определяются значения (,) и значения соответствующих критериев, характеризующих решения по составам партий данных.

На нижнем уровне принятия решений реализуется метод определения порядка обработки партий, предусматривающий добавление партий в последовательности () таким образом, чтобы обеспечить наиболее быстрое формирование комплектов всех типов. Поэтому для сформированного решения по порядкам обработки партий реализуется распределение результатов по комплектам по мере окончания выполнения операций с ними в конвейерной системе. С целью обеспеения оперативности формирования комплектов из результатов обработки и удовлетворения требования формирования комплектов в соответствии с их директивными сроками введены способы предварительного упорядочивания идентификаторов комплектов. Т.е. в зависимости от вида решаемой задачи (определения составов партий при условии оперативного формирования комплектов, определения составов партий при условии формирования комплектов с учетом заданных для них директивных сроков) идентификаторы типов комплектов *g*  () и номера  () комплектов соответствующих типов будут упорядочиваться различным образом.

Для первой задачи введены в рассмотрение следующие обозначения, которые позволят сформулировать метод упорядочивания идентификаторов комплектов с целью их последующего формирования из результатов обработки: – исходное множество идентификаторов типов комплектов, которые формируются из результатов обработки данных (первоначальлный вид множества ); – множество идентификаторов типов комплектов, упорядоченное в соответствии с рассмотренным ниже способом; – идентификатор типа комплекта, выделяемый из множества в соответствии с введенным в рассмотрение признаком и добавляемый в множество ; *h*- позиция идентификатора типа комплекта, на которой он размещается в множестве ; – идентификатор типа комплекта, занимающий в множестве *h*-ю позицию; – параметр, обратный относительной сложности формирования комплекта *g*-го типа, вычисляемый по формуле  (т.е. большее значсение параметра соответствует простому для формирования типу комплекта, меньшее значение соответствует сложному для формирования типу комплекта), – счетчик количетва результатов обработки данных *i’*-го типа, добавленных в -ый комплект *g’*-го типа (). Тогда, вводимый в рассмотрение способ реализует формирование упорядоченного множества типов комплектов на основе исходного множества .

Первый этап способ формирования упорядоченного множества типов комплектов имеет следующую последовательность шагов:

1) формирование исходного множества в виде: ;

2) задание исходного значения позиции *h* в множестве для рассматриваемого типа данных равной ();

3) среди типов данных выбирается такой тип данных , для которого выполняется следующее условие: ;

4) тип комплектов добавляется в множество на позицию *h* и исключается из множества : , ;

5) модификация значения *h* позиции типа данных в множестве : *h=h-1*; если выполняется условие , тогда реализуется переход на шаг 3;

6) в случае выполнена первоначальное упорядочивание типов комплектов *g* в множестве .

В результате реализации рассмотренной последовательности шагов типы комплектов *g* упорядочиваются в множестве в соответствии с значениями , т.е. множество определено в виде , где типы комплектов и размещаются на *h*-ой и (*h+1*)-ой позициях в этом множестве в том случае, если .

Описание алгоритма упорядочивания комплектов и распределения результатов по комплектам

После формирования с использованием рассмотренного способа первоначального вида множества , реализуется уточнение *h-*ых позиций типов комплектов при выполнении условия для подряд удущих идентификаторов и вида: . Уточнение (изменение) позиций *h* и (*h+1*) для типов комплектов и , для которых выполняется в соответствии со следующей последовательностью шагов:

1) в множестве определяются такие типы комплектов и , для которых выполняется условие ;

2) для определенных таким образом типов комплектов и выполняется вычисление значений параметров и ; в том случае, если для параметров и выполняется условие , тогда для рассматриваемых типов комплектов и производится обмен их позициями в множестве .

3) приведенные действия выполняеются для всех типов комплектов, для которых выполняется условие .

В итоге на основе множества вида формируется новое множество следующего вида .

Алгоритм определения моментов времени (,,), окончания формирования *i-*ых компонент во всех -ых комплектах всех *g*-ых типов (распределения результатов обработки партий по комплектам) в соответствии с расписанием , имеет следующий порядок шагов:

1) для *g-*ых типов комплектов () задание значения параметра равным 1 ();

2) для каждого *i-*го типа данных задание значения параметра , используемого при реализации алгоритма, равным 0 ();

3) задание индекса *i’*  типа данных, которые будут распределяться по комплектам *g*-ых типов (): , где *I* – множество типов данных, партии которых обрабатываются в системе; ;

4) задание идентификатора *g’* типа комплектов, в которые будут включаться данные *i’*-го типа: ; ; в соответствии с определенным таким образом типом комплекта *g’* реализуетсяидентификация значения количества комплектов этого типа;

5) задание значений счетчиков количества данных *i’*-го типа, добавленных в -ые комплекты *g’*-го типа, равными 0 (,);

6) определение для *i’*-го типа позиции *h* партии в последовательности , данные из которой будут включаться в -ый комплект *g’*-го типа, где *h –* порядковый номер позиции партии *i’*-го типа в такой, что при ;

7) если , то выполняется переход на шаг 8; если , то реализуется переход на шаг 9;

8) если , то выполняется модификация и : ; ; выполняется переход на шаг 7; если , то реализуется присваивание (где *h*-текущий номер партии данных *i’*-го, распределяемой по комплектам) и переход на шаг 6;

9) определение значения: (– элемент матрицы моментов времени окончания формирования *i-*ых компонент -ых комплектов *g’*-го типа), модификация индекса формируемого комплекта *g’*-го типа: ; если , то выполняется переход на шаг 7; если , то выполняется переход на шаг 10;

10) проверка условия ; в случае его выполнения – переход на шаг 11; при выполнении условия задание идентификатора вида комплекта *g’*, в который будут включаться данные *i’*-го типа: ; ; задание значений счетчиков количества данных *i’*-го типа, добавленных в -ые комплекты *g’*-го типа, равными 0 (,); переход на шаг 7;

11) проверка условия , при его выполнении реализуется переход на шаг 12; при выполнении условия осуществляется модификация идентификатора типа *i’*, данные которого распределяются по комплектам *g*-ых типов ():, ; реализуется переход на шаг 3;

12) останов алгоритма.

Значения элементов матриц () используются для определения моментов времени окончания формирования каждого -го комплекта () *g-*го типа и последующего вычисления значений критерия (5.5), характеризующих решения по составам партий на верхнем уровне иерархии.

Для второй задачи планирования обработки партий при условии формирования комплектов из результатов и задании для них директивных сроков в рассмотрение введены дополнительные обозначения, использование которых позволит сформулировать алгоритм упорядочивания идентификаторов комплектов: – идентификатор типа комплекта, рассматриваемого на текущем шаге алгоритма (добавляемый в множество ); *h*- позиция идентификатора типа комплекта, на которой он размещается в множестве ; – идентификатор типа комплекта, занимающий в множестве *h*-ю позицию, – множество типов данных, используемых при определении моментов времени окончания формирования комплектов, – тип данных, которые добавляются в комплект -го типа на текущем шаге алгоритма; – позиция партии данных *i’*-го типа в последовательностях (), из которой результаты обработки добавляются в комплект *g’*-го типа; – счетчик количетва результатов обработки данных *i’*-го типа, добавленных в комплект *g’*-го типа (), – вектор моментов времени окончания формирования *i*-ых компонент комплекта *g’*-го типа (*i-*ым элементом вектора является момент времени окончания формирования *i*-ой компоненты комплекта *g’-*го типа), – запаздывание с формированием комплекта рассматриваемого *g’-*го типа. После реализации алгоритма упорядочивания комплектов всех  типов множество имеет вид .

Вводимый в рассмотрение способ реализует формирование упорядоченного множества идентификаторов типов комплектов (при условии повторения идентификаторов типов комплектов в этом множестве) и соответствующего ему множества .

Способ формирования упорядоченных множеств и имеет следующую последовательность шагов:

1) задание исходного значения позиций *h* в множестве и в множестве для комплекта рассматриваемого типа и его директивного срока равным 1 ();

3) определение наличия в матрице не нулевых значений элементов ; при отсутствии в матрице не нулевых значений реализуется переход на шаг 6;

4) в случае наличия в матрице элементов , определение элемента , удовлетворяющего условию ;

5) идентификатор *g’* типа комплекта добавляется в множество  в *h-*ой позиции (), значение , соответствующее *g’-*му типу комплекта, добавляется в множество в *h*-ой позиции ;

5) присваивание рассматриваемому элементу матрицы нулевого значения (); переход на шаг 3;

6) окончание алгоритма.

В результате реализации рассмотренной последовательности шагов типы комплектов *g* упорядочиваются в множестве , множество определено в виде , где типы комплектов и размещаются на *h*-ой и (*h+1*)-ой позициях в этом множестве в том случае, если . Также перед реализацией алгоритма определения моментов времени окончания формирования комплектов множество *I* типов результатов обработки данных инициализуриется следующим: .

Алгоритм определения моментов времени () и определения в случае запаздываний с формированием комплектов *g*-ых типов в соответствии с решением по порядку обработки партий , имеет следующий порядок шагов:

1) инициализация значений элементов множества *I’*: ;

2) для каждого *i-*го типа данных задание значения параметра , используемого при реализации алгоритма, равным 0 (, );

3) определение в множестве типа комплекта , в *i-*ые компоненты которого реализуется распределение результатов обработки партий, и соответствующего ему значения директивного срока окончания его формирования:

, 

4) определение индекса *i’*  результатов обработки, из которых реализуется формирование *i’*-ой компоненты комплекта *g’*-го типа:: , где *I’* –множество типов результатов, из которых формируются комплекты; модификация *I’*:;

5) задание значения счетчика количества результатов *i’*-го типа, добавленных в формируемый комплект *g’*-го типа, равным 0 ();

6) определение для результатов *i’*-го типа позиции *h* партии в последовательностях (), данные из которой будут включаться в комплект *g’*-го типа, где *h –* порядковый номер позиции партии *i’*-го типа в такой, что  при (где – соответствующая компонента матрицы *R*);

7) если , то выполняется переход на шаг 8; если то выполняется переход на шаг 9;

8) если , то выполняется модификация и : ; ; выполняется переход на шаг 7; если , то реализуется присваивание (где *h*-текущий номер партии данных *i’*-го, распределяемой по комплектам) и переход на шаг 6;

9) определение значения: (– *i’*-ый элемент вектора моментов времени окончания формирования *i-*ых компонент комплекта *g’*-го типа);

10) проверка условия , в случае его выполнения результаты всех типов добавлены в формируемый комплект *g’*-го типа; выполняется расчет момента времени  окончания формирования комплекта *g’*-го типа и времени запаздывания с формированием комплектов: ; ; добавление полученного значения длительности интервала запаздывания в рассчитываемый критерий : ; реализуется переход на шаг 11; в случае выполнения условия реализуется переход на шаг 4;

11) проверка условия ; в случае его выполнения – переход на шаг 12; при выполнении условия реализуется инициализация значений элементов множества *I’*: ; переход на шаг 3;

12) останов алгоритма.