

УДК 681.51

С.В. Минухин

Харьковский национальный экономический университет, Харьков

ПОДХОДЫ К ОРГАНИЗАЦИИ ПЛАНИРОВАНИЯ РАСПРЕДЕЛЕНИЕМ РЕСУРСОВ В GRID-СИСТЕМАХ

Предложены подходы к организации планирования распределения ресурсов двухуровневой GRID-системе и процедуры и критерии оптимизации планирования распределения заданий в среде виртуальных сообществ GRID-систем. Предложенный критерий позволяет оптимизировать время выполнения отдельных заданий, а также суммарное время выполнения заданий входного потока, поступающего на двухуровневую GRID-систему. Для решения задач планирования предлагается использовать идеи рангового подхода к решению задач дискретной оптимизации.

Ключевые слова: GRID-система, брокер, виртуальное сообщество, двухуровневая архитектура, граф, задание, планирование, раскраска, ресурс.

Введение

Концепция построения и функционирования GRID зародилась в контексте важной, но относительно узкой проблемы построения сверхмощных вычислительных установок [1]. Сейчас исследуются возможности не только мощных вычислительных ресурсах метакомпьютинга: в качестве процессорных ресурсов (компонент, элементов) рассматриваются рабочие станции и персональные компьютеры (ПК).

Новое, современное понимание GRID-систем ассоциируется с понятием «ресурсы», включающем в себя все, что непосредственно или опосредовано участвует в компьютерной обработке данных. К ресурсам относят коммуникации, системы хранения и хранилища данных, информационные системы и т.д. Так, если приложение анализа данных понимать как ресурс, то это означает возможность удаленного доступа к нему без установки на компьютер пользователя. Выход за рамки задач высокопроизводительных вычислений выявляет реальное содержание

GRID как инфраструктуры для поддержки любой глобально распределенной обработки, для множества типов приложений: электронного бизнеса, распределенного производства, обработки и анализа данных. При этом не требуются высокопроизводительные коммуникации – в качестве коммуникационной среды может выступать Internet.

Известные подходы к построению GRID-инфраструктуры на основе находящихся в разных ее точках – кластерных узлах, в которые объединяются компьютеры, принадлежащие одному домену и соответствующие двухуровневой организации GRID-систем [1 – 3] – имеют тот недостаток, что в них предполагается, что компьютеры должны выделяться в GRID и могут быть использованы сотрудниками тех организаций, в которых они установлены, только частично. Для рассмотрения всей сети как единого компьютера, как показано в работах [4, 5], необходимо учитывать и рассматривать в задачах планирования использования ресурсов не только географическую, но и временную распределенность ресурсов.

Важным свойством систем GRID является то, что пользователю не нужно знать о физическом расположении ресурсов, отведенных для решения его задачи. Вся работа по управлению, перераспределению и оптимизации использования ресурсов ложится на «планировщика» – брокера ресурсов, и не регулируется пользователем. При этом для пользователя создается иллюзия работы в едином информационно-вычислительном пространстве, обладающем огромными вычислительными мощностями и объемом памяти. Для эффективной реализации такой возможности нужно предварительно определить принципы построения GRID-систем, в которых распределение заданий осуществляется в гетерогенной вычислительной среде, и решающие, в общем случае, такие задачи: определение функционального назначения каждой компоненты архитектуры GRID; определение общих принципов взаимодействия компонентов GRID; создание математического и программного обеспечения, гарантирующих эффективное и надежное функционирование GRID-систем.

Для этого в GRID-системе целесообразно использовать два уровня:

первый – серверы анализа состояний ресурсов и заданий клиентов, которые должны осуществлять сертификацию заданий клиентов, непосредственно подключенных к серверу, и сертификацию ресурсов, на которых возможно выполнить задания клиентов;

второй – управляющие серверы, которые содержат следующие службы:

информационная служба;

служба взаимодействия с исполнительными узлами;

служба поддержки пользователей;

служба диспетчеризации (распределения заданий);

служба мониторинга;

служба передачи данных;

служба работы с базами данных.

В программном обеспечении системы распределенных вычислений GRID выделяют пять основных компонент: компоненту адаптации ресурсов; компоненту связи; компоненту доступа к ресурсам; компоненту кооперации; компоненту координации [1, 7].

Компонента адаптации ресурсов является частью программного обеспечения GRID, которая работает на ресурсах и представляет их для использования вовне. **Компонента связи** объединяет протоколы коммуникации и безопасности с образованием унифицированной базы сетевых транзакций для вышележащих слоев. Протоколы коммуникации обеспечивают передачу данных, маршрутизацию и именование (стек TCP/IP: транспортный уровень (TCP, UDP), уровень Internet (IP, ICMP), прикладной уровень (DNS, OSPF, RSVP и т.д.). **Компонента доступа к ресурсам** определяет ряд протоколов и программных интерфейсов, которые определяют удаленное использование ресурсов GRID. С помощью средств этого слоя производится поиск ресур-

сов, дистанционная инициация, мониторинг и управление различными операциями. В отличие от кооперативного, слой доступа ограничен возможностью работы с индивидуальными ресурсами – без какого-либо учета глобального состояния GRID. В компоненте реализованы два типа протоколов: информационные и управляющие. Два информационных протокола базируются на протоколе LDAP. Службы первого из них – GRIP (GRID Resource Information Protocol) – устанавливаются на каждом ресурсе и собирают данные о его характеристиках (конфигурации, платформе) и состоянии (текущей загрузке). Информационная модель GRIP расширяема и позволяет, в принципе, представлять произвольные данные. Распределенная модель поддерживается вторым протоколом регистрации ресурсов GRRP (GRID Resource Registration Protocol), посредством которого сведения о наличии и местоположении GRIP сообщаются серверу GHS (GRID Index Information Server), на который впоследствии подкачиваются данные со всех зарегистрированных серверов GRIP. **Компонента кооперации** формируется над слоем удаленного доступа, позволяя, в отличие от последнего, взаимодействовать не с индивидуальным ресурсом, а с их совокупностью. На этом уровне GRID рассматривается уже как организованная среда. К этому уровню можно отнести следующее программное обеспечение: служба каталогов; служба брокеров; служба мониторинга и диагностики; служба репликации; служба авторизации; служба учета и платежей. **Компонента координации** представляет собой планировщик – брокер ресурсов – и требует не только решения внутренних задач – реализации алгоритмов планирования в глобальной распределенной среде, но и соответствующих расширений во всех слоях GRID-системы.

Компонента координации является наименее разработанной и наиболее важной с точки зрения реализации целей функционирования GRID и эффективности информационных систем. Следует отметить, что широко известная система Globus Toolkit не использует брокер ресурсов, оставляя задачу его реализации разработчикам, создающим программное обеспечение GRID. Поэтому в данной работе будут предложены и рассмотрены принципы организации функционирования данной компоненты, оптимизационные модели ее функционирования.

Цель исследования – разработка подходов и эффективных методов распределения вычислительных ресурсов и их координации в рамках системы GRID на основе принципов работы глобального и локального планировщиков (брокера ресурсов). К ним относятся: пользуясь информацией о состоянии GRID-системы, планировщик определяет наиболее подходящие ресурсы для выполнения конкретной задачи и резервирует их; во время выполнения задачи может запросить у планировщика дополнительные ресурсы или освободить избыточные; после завершения выполнения задачи все отведенные для

нее вычислительные ресурсы освобождаются, а ресурсы памяти могут быть использованы для хранения полученных результатов.

Важным свойством систем GRID является то, что пользователю не нужно знать о физическом расположении ресурсов, отведенных для решения его задачи. Вся работа по управлению, перераспределению и оптимизации использования ресурсов ложится на планировщика и выполняется незаметно для пользователя (свойство прозрачности системы). Автоматизация реализации таких функций является основной задачей рассматриваемой в данной работе проблемы.

Сформулируем принципы построения GRID-систем, в которых распределение заданий осуществляется в гетерогенной вычислительной среде, которые, в общем случае, должны обеспечивать решение следующих задач:

- определение функционального назначения компонент архитектуры GRID;
- определение общих принципов взаимодействия компонент GRID;
- создание математического и программного обеспечения, гарантирующего эффективное и надежное функционирование GRID-систем.

Для этого в GRID-системе предлагается использовать два основных уровня: первый уровень – это серверы анализа состояний ресурсов и заданий клиентов, которые должны осуществлять сертификацию заданий клиентов, непосредственно подключенных к серверу, и сертификацию ресурсов, на которых возможно выполнить задания клиентов; второй уровень, на котором находятся шлюзы, к которым подключаются серверы анализа состояний ресурсов и заданий клиентов, выполняющую функцию организации виртуальных сообществ, необходимых для выполнения потоков заданий, поступающих от серверов анализа состояний ресурсов и заданий клиентов и управляющих процессом их выполнения. Шлюзы обмениваются информацией о свободных ресурсах и сертифицированных заданиях и при этом планирование распределением ресурсов осуществляет только *один шлюз*, имеющий на данный момент это право. Порядок может быть установлен на основе технологий, используемых в компьютерных сетях распределенных систем в качестве методов доступа к среде передачи данных.

Постановка и формализация задачи

Входными данными для второго уровня системы являются множество Z сертифицированных задач в GRID и множество сертифицированных ресурсов Y . В результате сертификации каждую задачу характеризуют: объемом памяти, который она требует для ее реализации и требованиями к архитектуре ресурса, на котором она должна решаться; средним ожидаемым временем решения для наихудшего случая; договорной ценой ее решения. Каждый ресурс после сертификации характеризуют: интерва-

лом Δt_i времени, в течении которого данный ресурс может быть использован; характеристиками архитектуры и памяти ресурса; договорной ценой (стоимостью) предоставления ресурса. При этом будем полагать, что в подмножестве $z \in \{Z\}$ сертифицированных задач они могут решаться независимо друг от друга. Подмножество ресурсов $u \in \{Y\}$, предназначенных для решения подмножества задач $z \in \{Z\}$, образует виртуальное сообщество. Основной задачей является формирование принципов организации виртуальных сообществ для выполнения множества сертифицированных задач Z .

Будем полагать, что планирование выполнения заданий должно учитывать не только пространственное распределение ресурсов, но и временное распределение, задаваемое множеством интервалов времени $\{\Delta t_i\}$, в течении которого они выполняются. Для решения задачи планирования распределением задач по ресурсам внутри виртуальных сообществ предлагается разбить подмножества ресурсов с учетом их временной занятости на такие подмножества, в которых временные интервалы $\{\Delta t_i\}$ не пересекаются во времени. Последнее условие может быть реализовано на основе следующей формальной модели.

Отрезок времени $\Delta t_i = (t_k, t_q)$ определяет интервал времени, в течение которого можно задействовать некоторый произвольный j -й ресурс для выполнения заданий таким образом: с момента времени t_k ресурс для выполнения заданий задействуется и в момент времени t_q – освобождается. Таким образом, в общем случае, имеется p таких интервалов и требуется найти минимальное количество виртуальных сообществ, на которое можно разбить свободные ресурсы таким образом, чтобы в каждом сообществе все ресурсы могли независимо друг от друга выполнять требуемые задания. Для получения решения воспользуемся следующим графовым представлением. Для этого построим граф G , в котором каждой вершине графа i соответствует отрезок Δt_i , и соединим вершины i и j в графе G в том случае, если отрезки Δt_i и Δt_j пересекаются во времени.

Утверждение. Искомое количество сообществ равно хроматическому числу графа G .

Действительно, на всех ресурсах, имеющих одинаковый цвет, задания могут решаться независимо друг от друга. С другой стороны, когда имеется некоторое количество ресурсов для обслуживания всех заданий, то, окрасив одним цветом все отрезки, соответствующие вершинам графа G , получаем виртуальные сообщества ресурсов, в которых времена работы ресурсов не пересекаются во времени.

Следовательно, минимальное количество красок, с помощью которых можно раскрасить граф G , и определит минимальное количество виртуальных сообществ, в которых можно независимо выполнять некоторое подмножество заданий.

Оптимальная раскраска графов. Раскраска графов является одной из основных задач в теории графов и построении вычислительных систем и се-

тей, эффективное решение которой реализуется в различных приложениях [2].

Как показано в [2], задача раскраски графа может рассматриваться, как задача линейного булевого программирования. Решение этой задачи может быть осуществлено разработанными методами решения задач булевого программирования [3], но при этом из-за большой размерности ее целесообразно решать в графовой постановке, используя процедуру динамического программирования, предложенную в [2], на основе следующей теоремы.

Теорема 1. Если граф G является g -хроматическим, то он может быть раскрашен с использованием g (или меньшего числа) красок с помощью следующей процедуры: сначала в один цвет окрашивается некоторое максимально независимое множество $S_1[G]$, затем окрашивается в следующий цвет множество $S_2[\langle X - S_1[G] \rangle]$ и т.д. до тех пор, пока не будут раскрашены все вершины графа.

То, что такая раскраска, использующая только g цветов, всегда существует, может быть установлено следующим образом. Пусть существует раскраска в g цветов, такая, что одно или больше множеств, окрашенных в один и тот же цвет, не являются максимально независимыми множествами в смысле, упомянутом ранее. Пронумеруем цвета произвольным образом. Очевидно, что можно всегда покрасить в цвет 1 те вершины (пусть это множество вершин \bar{V}_1), которые не были окрашены в этот цвет, и которые образуют независимое максимальное множество вместе с множеством V_1 всех вершин графа, уже окрашенных в цвет 1. Эта новая раскраска возможна потому, что никакая вершина из множества \bar{V}_1 не является смежной ни с какой вершиной из V_1 , и, следовательно, всякая вершина, смежная хотя бы с одной вершиной из \bar{V}_1 , окрашена в цвет, отличный от цвета 1, и поэтому не затрагивается процедурой перемены цвета вершин из \bar{V}_1 .

Рассматривая теперь подграф $\langle X - V_1 \cup \bar{V}_2 \rangle$ и проводя с ним аналогичные манипуляции, окрасим в цвет 2 какое-то (новое) независимое максимальное множество и т.д. Раскраску указанного в теореме вида называют оптимальной независимой раскраской [2]. Тогда, используя известные эффективные алгоритмы определения минимального вершинного покрытия произвольного графа (обозначим его A_M) и процедуры динамического программирования, определяемых теоремой 1, можно получить алгоритмы раскраски произвольного графа.

Математические модели планирования

Формулирование математических моделей начнем с наиболее важной задачи распределения заданий в системе. Исходными данными для решения этой задачи являются выполняемые системой функций, которые могут быть формализованы в виде множества

решаемых заданий. Для каждого из заданий могут быть заданы возможные варианты его решения, и каждое задание может состоять из нескольких этапов (задач). Связи между заданиями и этапами (задачами) задаются в виде графа, ребра которого характеризуют соотношения следования, существующих решаемым заданиям и задачам. Кроме этого, задается множество ресурсов системы и связей между ними и виды технических средств, которые используются в системе. Требуется таким образом осуществить распределение заданий по ресурсам и уровням системы, при котором максимизируется эффект W от их выполнения.

В качестве критериев оптимизации предлагается использовать следующие частные критерии:

1. Минимизация затрат на реализацию заданий в системе

$$\min \sum_{i=1}^I \sum_{j=1}^J W_{ij} X_{ij}, \quad (1)$$

где $i = (\bar{1}, \bar{I})$ – множество заданий, реализуемых в системе; $j = (\bar{1}, \bar{J})$ – множество обслуживающих ресурсов системы; W_{ij} – затраты на реализацию i -го задания в j -м ресурсе;

$$X_{ij} = \begin{cases} 1, & \text{если задание } i \text{ выполняется в } j\text{-м ресурсе;} \\ 0, & \text{в противном случае.} \end{cases}$$

2. Минимизация общего времени выполнения всех задания системы

$$\min \sum_{i=1}^I \sum_{j=1}^J t_{ij} X_{ij}, \quad (2)$$

где t_{ij} – время выполнения i -го задания j -м ресурсом.

3. Минимизация максимального времени выполнения задания в системе

$$\min \left\{ \max_j \sum_{i=1}^I t_{ij} X_{ij} \right\}. \quad (3)$$

Возможна оптимизация по более сложным критериям, таким как, обеспечение требуемого времени готовности системы и др.

Качественные ограничения в частных задачах оптимизации используются следующие:

1. Связи между заданиями, т.е. задан граф G_F .

2. Связи между узлами, т.е. задан граф G_M .

3. Общие затраты на реализацию задания в системе:

$$\sum_{i=1}^I \sum_{j=1}^J W_{ij} X_{ij} \leq W_{\text{доп}}. \quad (4)$$

4. Затраты на реализацию задания в ресурсах:

$$\sum_{i=1}^I W_{ij} X_{ij} \leq W_{j\text{доп}}, \quad j = (\bar{1}, \bar{J}). \quad (5)$$

5. Загрузка каждого из ресурсов:

$$\sum_{i=1}^I \lambda_i t_{ij} X_{ij} \leq \rho_j, \quad j = (\bar{1}, \bar{J}), \quad (6)$$

где λ_i – интенсивность поступления i -ого задания на решение.

6. Общее время выполнения задания:

$$\sum_{i=1}^I \sum_{j=1}^J t_{ij} x_{ij} \leq T. \quad (7)$$

7. Время выполнения отдельных заданий:

$$\sum_{j=1}^J t_{ij} x_{ij} \leq \tau_i, \quad i = \overline{(1, I)}. \quad (8)$$

Рассмотрим основные частные задачи оптимизации.

Первая частная задача оптимизации процесса распределения заданий

Необходимо распределить i заданий $i = \overline{(1, I)}$ между j ресурсами $j = \overline{(1, J)}$, чтобы обеспечить минимум общих затрат (7) или минимум общего времени их выполнения (8) при ограничениях на загрузку каждого из ресурсов (6) или затраты в каждом j -м ресурсе (5). Математическая модель этой задачи может быть записана следующим образом:

найти

$$\min \sum_{i=1}^I \alpha_{ij} x_{ij} \quad (9)$$

при ограничениях:

$$\sum_{i=1}^I \beta_{ij} x_{ij} \leq b_j, \quad j = \overline{(1, J)}; \quad (10)$$

$$\sum_{j=1}^J x_{ij} = 1, \quad i = \overline{(1, I)}, \quad x_{ij} \in \{0, 1\}, \quad (11)$$

где α_{ij} – затраты (время выполнения) i -го задания в j -м ресурсе; b_j – допустимые затраты (загрузка) в j -м ресурсе;

$$x_{ij} = \begin{cases} 1, & \text{если задание } i \text{ выполняется в } j\text{-м ресурсе;} \\ 0 & \text{– в противном случае.} \end{cases}$$

Ограничение (11) означает, что каждая задание выполняется только одним ресурсом.

Вторая частная задача оптимизации процесса распределения заданий

Необходимо так распределить i заданий $i = \overline{(1, I)}$ между j ресурсами, $j = \overline{(1, J)}$, чтобы обеспечить минимум времени (общих затрат) (7) или минимум общего времени выполнения (8) при выполнении ограничений на общее время выполнения (7) или общие затраты (10).

Математическая модель этой задачи формулируется так: найти

$$\min \sum_{i=1}^I \sum_{j=1}^J \alpha_{ij} x_{ij} \quad (12)$$

при ограничениях:

$$\min \sum_{i=1}^I \sum_{j=1}^J \beta_{ij} x_{ij} \leq B; \quad (13)$$

$$\sum_{j=1}^J x_{ij} = 1, \quad i = \overline{(1, I)}, \quad x_{ij} \in \{0, 1\}, \quad (14)$$

где α_{ij} – затраты (время выполнения) i -го задания на j -м ресурсе; β_{ij} – время выполнения (затраты) i -го задания на j -м ресурсе; B – общее время выполнения (затраты) всех заданий.

Третья частная задача оптимизации процесса распределения заданий

Необходимо так распределить i заданий $i = \overline{(1, I)}$ между j узлами, $j = \overline{(1, J)}$, чтобы обеспечить минимум времени общих затрат (7) или минимум общего времени выполнения (8) при выполнении ограничений на общее время выполнения (7) и загрузку ресурсов (6), либо на общие затраты (10) и загрузку ресурсов (11). Математическая модель этой задачи имеет следующий вид:

найти

$$\min \sum_{i=1}^I \sum_{j=1}^J \alpha_{ij} x_{ij} \quad (15)$$

при ограничениях

$$\min \sum_{i=1}^I \sum_{j=1}^J \beta_{ij} x_{ij} \leq B; \quad (16)$$

$$\sum_{i=1}^I \sum_{j=1}^J c_{ij} x_{ij} \leq C_j, \quad j = \overline{(1, J)}; \quad (17)$$

$$\sum_{j=1}^J x_{ij} = 1, \quad i = \overline{(1, I)}, \quad x_{ij} \in \{0, 1\}. \quad (18)$$

Рассмотренные постановки задач являются основными при оптимизации процессов распределения заданий в GRID-системах.

Выводы

Рассмотренный подход позволяет автоматизировать решение задач распределения ресурсов как с учетом временных требований к выполнению заданий, так и с учетом затратных (стоимостных) ограничений, возникающих в процессе управления выполнением заданиями в двухуровневой системе. Преимущество планирования при таком подходе заключается в том, что время жизни всех виртуальных сообществ в этом случае будет определяться суммой отрезков времени множества $\{\Delta t_i\}$, соответствующим времени использования ресурсов, вошедших в виртуальное сообщество. При этом упрощается сам процесс планирования, поскольку задача о назначениях полиномиально разрешима, что позволяет управлять заданиями в реальном времени. В случаях, когда в виртуальных сообществах на данный момент времени отсутствуют необходимые ресурсы, возможно перераспределение заданий внутри сообщества с целью максимизации эффекта выполнения заданий, что обеспечивает устойчивое функционирование информационной системы. Для решения задач дискретной оптимизации, возникающих в процессе перераспределения задач внутри виртуальных сообществ, можно применить эффективные методы решения, предложенные в работах [5 – 7], основанные

на идеях рангового подхода к их решению, и позволяющие решать их с меньшей трудоемкостью и большей точностью в масштабе реального времени.

Список литературы

1. Коваленко В.Н. Организация ресурсов грид. Препринт №63 / В.Н. Коваленко, Д.А. Корягин. – М.: ИПМ РАН им. М.В. Келдыша, 2004. – 25 с.
2. Кристофидес Н. Теория графов. Алгоритмический подход / Н. Кристофидес. – М.: Мир, 1978. – 309 с.
3. Калинин В.Н. Теория систем и оптимального управления. Часть 2. Понятия, модели и алгоритмы оптимального выбора / В.Н. Калинин, Б.А. Резников, Е.И. Варакин. – М.: МО СССР, 1987. – 590 с.
4. Листровой С.В. Метод решения задачи определения минимальных вершинных покрытий и независимых максимальных множеств / С.В. Листровой, С.В. Яблочков // Электронное моделирование. – 2003. – Т. 25, № 2. – С. 31-40.

5. Listrovoy S.V. Solution method on the basis of rank approach for integer linear problems with boolean variables / S.V. Listrovoy, D.Yu. Golubnichiy, E.S. Listrovaya // Engineering Simulation. – 1999. – Vol. 16. – P. 707-725.

6. Методы моделирования и дискретной оптимизации вычислительных систем реального времени / В.Я. Жихарев, В.М. Илюшко, Л.Г. Кравец, С.В. Листровой, В.С. Харченко; под ред. В.Я. Жихарева. – Харьков – Житомир: ЖГУ, 2004. – 494 с.

7. Методы и модели планирования ресурсов в GRID-системах / В.С. Пономаренко, С.В. Листровой, С.В. Минухин, С.В. Знахур. – Х.: ИД «ИНЖЭК», 2008. – 408 с.

Поступила в редколлегию 21.04.2009

Рецензент: д-р техн. наук, проф. С.В. Листровой, Харьковская государственная академия железнодорожного транспорта, Харьков.

ПІДХОДИ ДО ОРГАНІЗАЦІЇ ПЛАНУВАННЯ РОЗПОДІЛОМ РЕСУРСІВ В GRID-СИСТЕМАХ

С.В. Мінухін

Запропоновані підходи до організації планування розподілу ресурсів дворівневій GRID-системі і процедури і критерії оптимізації планування розподілу завдань в середовищі віртуальних співтовариств GRID-систем. Запропонований критерій дозволяє оптимізувати час виконання окремих завдань, а також сумарний час виконання завдань вхідного потоку, що поступає на дворівневу GRID-систему. Для вирішення завдань планування пропонується використовувати ідеї рангового підходу до рішення задач дискретної оптимізації.

Ключові слова: GRID-система, брокер, віртуальне співтовариство, дворівнева архітектура, граф, завдання, планування, розфарбовування, ресурс.

ON THE GOING NEAR ORGANIZATION OF PLANNING ALLOCATION OF GRID RESOURCES

S.V. Minukhin

Approaches are offered to organization of planning of allocation of resources to the two-tier GRID-system and procedures and criteria of optimization of planning of distributing of tasks in the environment of virtual associations of the GRID-systems. The offered criterion allows to optimize time of processing of separate jobs, and also total time of processing of jobs of input stream, acting on the two-tier GRID-system. For the decision of planning tasks it is suggested to use the ideas of grade approach to the decision of tasks of discrete optimization.

Keywords: GRID-system, broker, virtual association, two-tier architecture, count, task, planning, coloration, resource.