

Козловский Александр Львович

**МОДЕЛИ, МЕТОДЫ И АЛГОРИТМЫ РАСПРЕДЕЛЕНИЯ РЕСУРСОВ
ВИРТУАЛИЗОВАННЫХ ВЫЧИСЛИТЕЛЬНЫХ КЛАСТЕРОВ**

Специальности:

05.13.05 – «Элементы и устройства вычислительной техники и систем управления»

05.13.01 – «Системный анализ, управление и обработка информации (приборостроение)»

АВТОРЕФЕРАТ

диссертации на соискание ученой степени
кандидата технических наук

МОСКВА

2012

Работа выполнена в ФГУП «Московское орденов Октябрьской Революции и
Трудового Красного Знамени конструкторское бюро «Электрон»

Научный руководитель: Доктор технических наук, с.н.с.
Гольдин Виктор Вольфович

Официальные оппоненты: Доктор технических наук, профессор
Увайсов Сайгид Увайсович

Кандидат технических наук, доцент
Штейнберг Виталий Иосифович

Ведущая организация: ОАО «Научно-исследовательский институт
вычислительных комплексов им. М.А. Карцева»

Защита состоится 16 февраля 2012 года в _____ часов на заседании
диссертационного совета Д 217.047.01 при ФГУП «Научно-исследовательский и
экспериментальный институт автомобильной электроники и электрооборудования»
по адресу: 105187, г. Москва, ул. Кирпичная д. 39/41.

С диссертацией можно ознакомиться в библиотеке ФГУП «Научно-
исследовательский и экспериментальный институт автомобильной электроники и
электрооборудования»

Автореферат разослан « ____ » января 2012 г.

Ученый секретарь диссертационного
совета Д 217.047.01
доктор технических наук, с.н.с.



Варламов О. О.

ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

Актуальность темы.

На сегодняшний день виртуализация серверной инфраструктуры является одной из основных тенденций развития отрасли информационных и коммуникационных технологий. Многие ведущие мировые производители программного обеспечения выпускают на рынок соответствующие продукты. Столь широкое внедрение технологий виртуализации связано с тем, что в последнее время был совершен большой технологический прорыв в области виртуализации операционных систем, открывший огромные возможности и перспективы. Под виртуализацией операционных систем понимают процесс создания на физическом сервере или кластере так называемой виртуальной машины, в которой устанавливается своя собственная операционная система. Виртуальных машин на одной физической платформе может быть множество, при этом каждая виртуальная машина имеет свои собственные виртуальные аппаратные компоненты: память, процессор, жесткий диск, сетевые адаптеры. Эти ресурсы резервируются виртуальной машиной за счет физических ресурсов аппаратного обеспечения сервера или кластера. Такая модель организации вычислительных систем впервые появилась еще в 70-х годах прошлого века в мэйнфреймах корпорации IBM System 360/370, когда требовалось сохранить предыдущие версии экземпляров операционных систем. Но лишь относительно недавно эта технология стала широко доступна для современных серверных систем.

В связи с увеличением вычислительной сложности задач управления, решаемых с помощью виртуализованных кластеров, растет потребность в сбалансированном распределении вычислительных мощностей аппаратного обеспечения для работы различных приложений. Следовательно, необходима разработка методик, моделей и алгоритмов оптимизации распределения ресурсов при виртуализации серверной и коммуникационной инфраструктуры локальной вычислительной сети.

Объектом исследований являются: унаследованные и создаваемые серверные парки, состоящие из множества узлов, связанных локальной вычислительной сетью,

технология кластерной виртуализации, применяемая для более эффективного использования вычислительных ресурсов.

Предмет исследований: методы, модели, алгоритмы, программно-аппаратные средства, обеспечивающие оптимальное распределение ресурсов вычислительных кластеров, принимая во внимание надежность характеристики их элементов и стоимость дальнейшей поддержки системы в целом.

Цель и задачи диссертационного исследования.

Целью работы является повышение эффективности использования ресурсов виртуализованных вычислительных кластерных систем, обеспечивающих надежность функционирования и минимизацию стоимости обслуживания в ходе эксплуатации.

Для реализации поставленной цели необходимо решение следующих задач:

1. Системный анализ существующих подходов к решению проблемы повышения эффективности использования вычислительных ресурсов. Анализ существующих моделей распределения ресурсов, используемых в различных областях деятельности.
2. Анализ существующих методик и технологий получения характеристик производительности устройств вычислительной техники, а также требований к вычислительным ресурсам программного обеспечения.
3. Разработка методики реорганизации серверного парка с применением технологии виртуализации для повышения эффективности использования вычислительных ресурсов.
4. Синтез моделей надежности элементов и устройств вычислительной техники, учитывая фактор снижения надежности аппаратного обеспечения при увеличении его загрузки из-за повышения температуры электрорадиоизделий, входящих в состав данной системы.
5. Разработка математических моделей, алгоритмов и методов решения задачи оптимального планирования распределения вычислительных ресурсов.
6. Разработка программной реализации созданных алгоритмов с графическим интерфейсом, которая бы наглядно демонстрировала на основании входных

данных схему распределения виртуальных машин внутри физического кластера.

7. Проведение экспериментальной проверки разработанной методики распределения ресурсов.

Методы исследования.

В работе используются методы математического моделирования, методы оптимизации, математической статистики и теории вероятностей, динамического программирования. При разработке программного комплекса использованы современные методы создания программного обеспечения.

Научная новизна результатов, выносимых на защиту.

При решении задач, поставленных в диссертационной работе, получены следующие новые научные результаты:

1. Разработаны математические модели и алгоритмы распределения ресурсов вычислительных кластеров, которые, в отличие от известных, позволяют генерировать необходимое число различных конфигураций системы, контролируя, в том числе, надежность характеристики оборудования (тепловой режим работы, наличие систем резервирования и пр.).
2. Предложен новый универсальный подход к оценке планирования ресурсов виртуализованного кластера, а также его программная реализация, отличающиеся от известных тем, что позволяют эффективно планировать распределение вычислительных мощностей как унаследованных, так и новых программно-аппаратных комплексов. Данный метод не имеет ограничений на размер анализируемого комплекса и не зависит от производителя программного или аппаратного обеспечения.
3. Создана методика планирования распределения вычислительных ресурсов, позволяющая динамически определять конфигурацию системы с минимальной стоимостью обслуживания, которая в отличие от традиционной методики, основанной на приближенном подходе, базируется на методах математического моделирования и измерении физических величин.

Практическая значимость результатов работы состоит в том, что предложенные модели, алгоритмы, программные средства и методическое обеспечение позволяют автоматизировать процесс принятия решений о распределении ресурсов виртуализованных вычислительных кластеров и, соответственно, повысить надежность и эффективность использования программно-аппаратного комплекса в целом.

Результаты, выносимые на защиту:

1. Математические модели и алгоритмы распределения ресурсов виртуализованных вычислительных кластеров.
2. Метод оптимального планирования распределения вычислительных ресурсов, позволяющий синтезировать конфигурацию системы с минимальной стоимостью обслуживания и высокими надежностными характеристиками.
3. Методика автоматизированной обработки данных о производительности аппаратных средств и требования к ресурсам приложений, с последующей генерацией необходимого числа вариантов распределения вычислительных ресурсов.

Апробация результатов работы. Основные положения диссертационной работы доложены и обсуждены на Международных научно-практических конференциях «Инфо-2010», (г. Сочи, 2010) и «Инфо-2011» (г. Сочи, 2011), а также на 19-ой Международной научно-технической конференции «Современное телевидение и радиоэлектроника» (г. Москва, 2011).

Публикации. По материалам работы опубликовано 5 научных работ, в том числе 2 работы – в журналах из перечня ВАК. Получено свидетельство об официальной регистрации программы для ЭВМ.

Реализация и внедрение результатов работы.

Основные результаты диссертационной работы внедрены в ФГУП «МКБ Электрон», ООО НПЦ «Квадра», а также в учебный процесс Московского государственного текстильного университета имени А.Н.Косыгина по дисциплине «Теоретические основы автоматизированного управления» в лекционном курсе и при выполнении лабораторных работ.

Структура и объем работы. Диссертационная работа состоит из введения, четырех глав с выводами, заключения, списка литературных источников и приложений. Работа изложена на 163 страницах машинописного текста. Список литературных источников включает 76 наименований.

КРАТКОЕ СОДЕРЖАНИЕ РАБОТЫ

ВО ВВЕДЕНИИ обосновывается актуальность темы диссертации, охарактеризовано состояние исследуемых вопросов, приводится общая характеристика предметной области, определяются цель и задачи исследования. Сформулированы научная новизна и практическая значимость полученных результатов. Представлены состав и краткое описание работы, приведены сведения об апробации работы и публикациях.

В ПЕРВОЙ ГЛАВЕ осуществлен анализ используемых на практике технологий виртуализации серверной и коммуникационной инфраструктуры, произведено их сравнение. Приведены преимущества и недостатки технологий, описаны предпосылки перехода на виртуализованную инфраструктуру, указаны основные факторы неудачи проектов по виртуализации серверной инфраструктуры.

Для того чтобы физический сервер работал как несколько виртуальных серверов необходимо, чтобы характеристики его аппаратной части воссоздавались с помощью программного обеспечения. Это достигается особым методом построения программного обеспечения, называемым абстракцией.

Такое программное обеспечение (ПО) используется во многих системах включая системы семейства Microsoft Windows. Windows Hardware Abstraction Layer (HAL) - типичный пример абстракции. Windows HAL обеспечивает взаимодействие всех драйверов и ПО с аппаратной частью в стандартизированном формате. Это упрощает написание ПО и драйверов, т.к. разработчикам не нужно создавать отдельное ПО для каждой модели компьютера в отдельности. Абстракция в терминах виртуализации – это набор стандартных аппаратных устройств, целиком управляемых программным обеспечением. Таким образом, это ПО, которое выглядит и действует как аппаратное обеспечение. Технология виртуализации позволяет установить операционную систему на аппаратные средства, которые на

самом деле не существуют. Достигается возможность с помощью виртуализации разделить ресурсы одного физического сервера для одновременной работы разных вычислительных сред. Причем разные вычислительные среды могут как взаимодействовать, так и быть полностью отделены друг от друга. Кроме того, каждая из вычислительных сред даже не будет подозревать, что она существует в общей виртуальной среде. Такие вычислительные среды принято называть виртуальными машинами (VM). Практически всегда каждая такая VM содержит в себе некие операционные системы (Linux, Windows, Netware и т.д.), называемые гостевыми операционными системами. Инструкции для VM обычно передаются физическому аппаратному обеспечению, что дает возможность вычислительной среде работать быстрее и более эффективно, чем эмуляции, хотя более сложные инструкции должны быть перехвачены и интерпретированы, чтобы обеспечить совместимость и абстракцию с физическим аппаратным обеспечением. На Рисунке 1 представлена схема работы виртуализированного сервера.

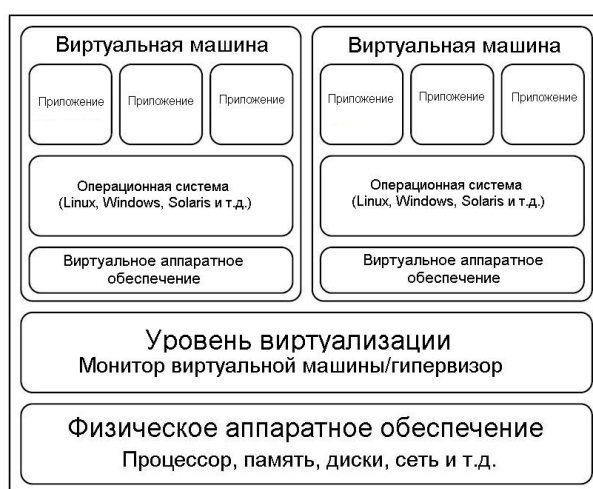


Рисунок 1. Виртуализованная архитектура

Приведен обзор моделей распределения ресурсов для решения задач планирования. Показано, что в настоящее время проблема эффективного распределения ресурсов (экономических, транспортных, вычислительных и т.д.) требует своего решения в самых разных сферах деятельности человека. Существует множество готовых моделей оптимального распределения ресурсов, направленных

для решения различных частных задач, как чисто экономических, так и технических. К сожалению, какой-либо универсальной модели эффективного распределения ресурсов пока нет, поэтому для каждой конкретной ситуации необходимо разрабатывать новую модель/алгоритм.

Рассматриваются наиболее известные модели распределения ресурсов и приводится анализ возможности их применения для решения специфичной задачи нахождения оптимального распределения определенного количества виртуальных машин по физическим серверам.

ВТОРАЯ ГЛАВА посвящена разработанным в рамках диссертации математическим моделям оптимизации загрузки серверов для обеспечения надежности и контроля функционирования программной и аппаратной части устройств вычислительной техники, включенных в состав виртуализованного кластера. В качестве оптимизируемых параметров естественным представляется максимизация уровня загрузки серверов и минимизация их стоимости в денежном выражении.

Полученные модели статического распределения серверов соответствуют канонической задаче распределения ресурсов, поскольку отражают процедуру соответствия виртуальных машин минимальному числу физических серверов.

Принимаются следующие предположения: есть n сервисов/приложений $s_1, s_2, \dots, s_j, \dots, s_n (j \in J)$, которые установлены на m серверах $x_1, x_2, \dots, x_i, \dots, x_m (i \in I)$. Различные виды ресурсов, такие как память, процессор и т.д. обозначим как $r_1, r_2, \dots, r_k (k \in K)$. Для каждого сервиса s_j должно быть назначено u_{jk} условных единиц ресурса r_k . Каждый физический сервер имеет статическую величину производительности s_{ik} ресурса r_k . Определим y_i как двоичную переменную, показывающую какие серверы используются, c_i - показывающую приблизительную стоимость сервера, а x_{ij} - указывающую на то, какое приложение работает на конкретном сервере. Принимая во внимание наличие различных типов ресурсов

(память, частота процессора и др.), оптимальное решение задачи может быть сформулировано следующим образом:

$$\min \sum_{i=1}^m c_i y_i$$

где:

$$\sum_{i=1}^m x_{ij} = 1, \forall j \in J; \sum_{j=1}^n u_{jk} x_{ij} \leq s_{ik} y_i, \forall i \in I; \forall k \in K \quad (1)$$

$$y_i, x_{ij} \in \{0,1\}, \forall i \in I, \forall j \in J \quad (2)$$

Целевая функция минимизирует стоимость серверов, в то время как набор ограничений (1) обеспечивает разовое распределение каждого сервиса, а набор ограничений (2) обеспечивает непревышение суммарной нагрузки нескольких приложений возможностям каждого физического сервера.

Полученная модель представляет собой потребность в ресурсах единичного сервиса как константу, неизменяемую со временем. Однако на практике характерны ситуации изменения нагрузки (потребности в ресурсах) сервиса во времени. Поэтому необходимо провести корректировку модели. Определим время T как индексированный набор интервалов $t = \{1, \dots, \tau\}$. Циклическую с течением времени нагрузку представим в виде матрицы u_{jkt} , описывающую объем производительности, требуемый сервисом s_j от типа ресурса r_k в течение интервала t . В соответствии с этой матрицей набор ограничений может быть задан как:

$$\sum_{j=1}^n u_{jkt} x_{ij} \leq s_{ik} y_i, \quad \forall i \in I, \forall j \in J, \forall t \in T$$

Возможность сокращения требуемого количества физических серверов является одним из главных аргументов в пользу выбора технологий виртуализации. Предложенная модель позволяет определить количественно, можно ли сэкономить физические ресурсы на основе статистических данных. Коэффициенты u_{jkt} зависят от характеристик загрузки серверов, которую требуется повышать, чтобы

минимизировать число физических серверов. Естественно предположить, что $u_{jkt} \leq s_{ik}$.

В данной модели учитываются в явном виде параметры серверов. Их реальные значения могут как оставаться неизменными, так и меняться, например, при замене серверов. В этом случае задание новых значений параметров должно являться результатом тестирования оборудования и исследования его производительности.

Далее анализируется взаимосвязь загрузки серверов с их надежностными показателями поскольку, как уже было упомянуто ранее, главной целью оптимизации распределения физических серверов между сервисами является балансировка коэффициента их загрузки и, следовательно, сокращение времени простоя. Однако необходимо учитывать тот факт, что электронные компоненты сервера при полной загрузке в процессе обработки информации работают в режиме, весьма близком к верхней границе допустимого интервала температур. Хорошо известно о нелинейной зависимости надежности электронных компонентов от температурного режима. Поэтому необходимо иметь в виду, что очевидно имеется некоторое пороговое значение коэффициента загрузки сервера, выше которого подниматься не имеет смысла, поскольку увеличение частоты отказов не только способствует росту затрат на резервное оборудование, но и будет приводить к снижению эффективности вычислительного процесса и даже к его нарушениям.

Таким образом, при построении модели виртуализованного кластера необходимо учитывать возможность отказа оборудования при повышении температуры компонентов системы. Выход некоторых компонентов из строя может привести к сбою всей системы в целом. При планировании консолидации серверного парка необходимо учитывать параметры надежности оборудования и заранее заботиться о «горячем резерве» компонентов.

ТРЕТЬЯ ГЛАВА посвящена созданию эффективного алгоритма практической реализации модели, предложенной во второй главе. Рассматривается известная задача о рюкзаке и производится ее модификация под условия модели. Кроме того,

рассматриваются приемы программирования для реализации полученного алгоритма.

На Рисунке 2 изображена блок-схема алгоритма модифицированной задачи о рюкзаке.

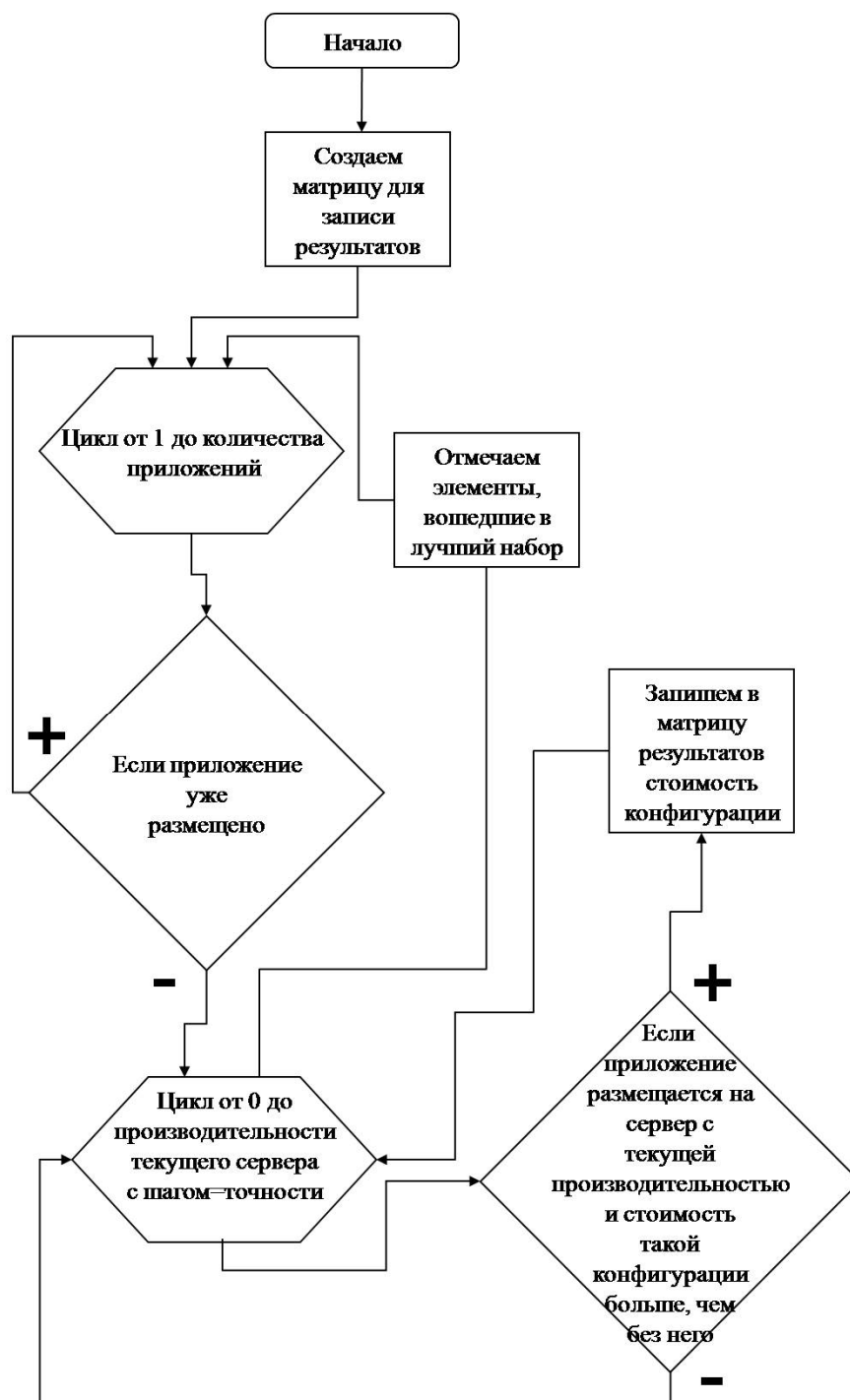


Рисунок 2. Блок-схема алгоритма модифицированной задачи о рюкзаке

Алгоритм решения модифицированной задачи о рюкзаке состоит в следующем:

1. Запоминаем производительность текущего сервера. Создаем матрицу A размерности $n+1$ на $(w/z)+1$, где n – количество приложений, w – вес рюкзака. z – это показатель точности алгоритма, то есть с таким шагом будет решаться задача.
2. Заполняем нулями начальную строку матрицы, так как она соответствует случаю, когда доступно 0 приложений.
3. Начинаем цикл по переменной p по количеству приложений от 1 до n .
4. Если приложение уже использовалось (то есть в массиве индикаторов занятости оно помечено как распределенное), то переходим к пункту 3.
5. Начинаем вложенный цикл по переменной j от 0 до w с шагом z . Здесь мы перебираем все возможные размеры рюкзака в пределах от 0 до w .
6. Присваиваем элементу $A[p, j/z]$ значение $A[p-1, j/z]$. Такое действие необходимо для того, чтобы на следующем шаге выбрать между этим значением и значением, описанным ниже.
7. Если теперь текущее приложение с номером p помещается в сервер с текущим объемом j , и при этом $A[p-1, j-wp]+1 > A[p, j]$ то текущему значению $A[p, j]$ присваиваем $A[p-1, j-wp]+1$. Где wp – это вес текущего приложения с номером p .
8. Таким образом, в пунктах 6 и 7 проверяются следующие данные: если приложение p не попало в сервер веса j , то такой сервер будет укомплектован только из приложений с номерами от 1 до $p-1$. Если же в сервер включен предмет p , то объем оставшихся приложений не превышает $j-wp$, а от добавления приложения p количество приложений увеличивается на 1. Из двух таких вариантов комплектования сервера нужно выбрать наилучший, то есть тот, в котором поместилось больше приложений.
9. Возвращаемся к пункту 5.

10. После завершения вложенного цикла сохраняем номер текущего приложения в переменную L и переходим к пункту 3. Сохранение номера необходимо, так как не все приложения могут быть свободны на момент действия процедуры, а этот номер необходим для вывода решения задачи.
11. После завершения внешнего цикла в ячейке $A[L, w/z]$ хранится ответ: максимальное количество приложений из числа доступных, которые могут быть размещены на текущем сервере.
12. Далее отметим приложения из доступного количества p (в начале работы процедуры их L), которые вошли в лучший комплект для сервера с объемом j (начинаем со значения w , то есть производительности текущего сервера). Процедура рекурсивная.
13. Если $A[p, j] = 0$, значит приложения требуют большей производительности чем возможно, программа ничего не отмечает и переходит к пункту 15, иначе переходит к пункту 13.
14. Если $A[p-1, j] = A[p, j]$, значит максимально укомплектовать сервер можно и без приложения p , переходим к пункту 11 с количеством доступных приложений $p-1$, иначе переходим к пункту 14.
15. Если же $A[p-1, j] \neq A[p, j]$, значит без текущего приложения максимально укомплектовать сервер нельзя, отмечаем это приложение, как использованное и переходим к пункту 11 со значением количества доступных приложений $p-1$ и с максимальной производительностью сервера $j - wp$, где wp – вес отмеченного только что приложения.
16. После выполнения рекурсивной процедуры задача полностью решена, так как мы отметили все необходимые приложения и посчитали их количество.

На Рисунке 3 изображена блок-схема общего алгоритма модели распределения ресурсов, в который встраивается полученный алгоритм модифицированной задачи о рюкзаке. Перед выполнением алгоритма «лучшая» стоимость полагается равной максимально возможной константе.

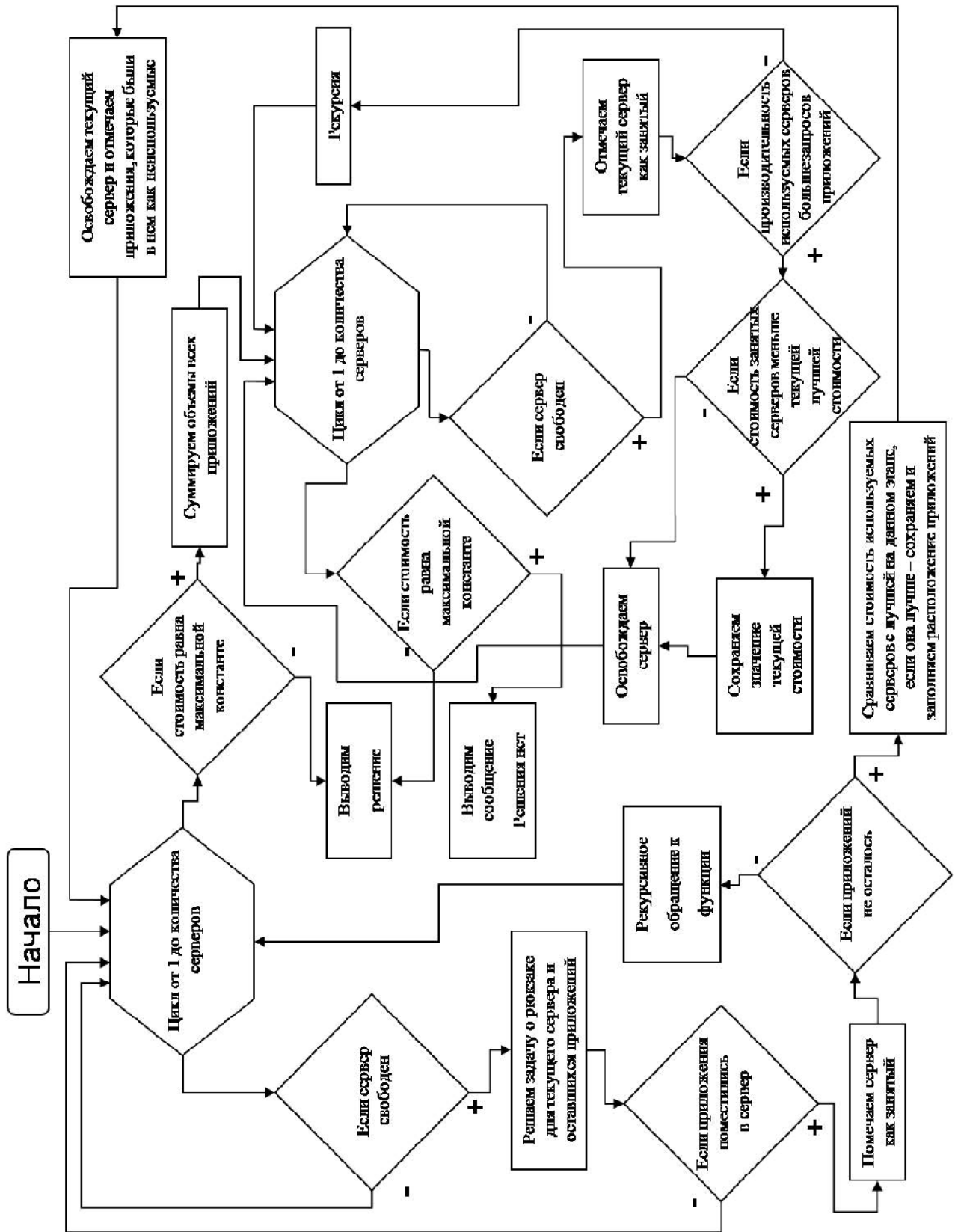


Рисунок 3. Блок-схема общего алгоритма решения задачи

1. Будем перебирать все возможные варианты размещения по серверам следующим образом. Начнем цикл от 1 до количества серверов. На i -м (текущем) шаге проверим, не занят ли сервер (для этого в программе необходимо предусмотреть массив индикаторов занятости серверов). Если сервер занят, вернемся к шагу 1.
2. Если сервер свободен, и не все приложения распределены по серверам (для приложений также необходимо создать массив индикаторов занятости), то для оставшегося набора предметов и текущего контейнера решается модифицированная задача о рюкзаке. В результате ее решения в массиве занятости приложений будут отмечены (как размещенные) приложения из числа тех, поместившихся на текущий сервер, что не были размещены до сих пор.
3. После этого сервер исключается из рассмотрения и помечается, как использованный. Если приложений не осталось (то есть мы разложили их по серверам), переходим к шагу 4, иначе – к шагу 6.
4. Стоимость всех помеченных как занятые на данный момент контейнеры – это стоимость данного варианта размещения. Если она меньше текущей «лучшей» стоимости, то данный набор отмечается как наилучший, стоимость запоминается как наилучшая, а конфигурация размещения сохраняется в специальном массиве.
5. Производится возвращение процедуры решения на шаг назад. То есть мы освобождаем текущий сервер от предполагаемых для него приложений, отмечаем приложения, как неиспользованные, отмечаем сервер, как незанятый и переходим к шагу 1 (не рекурсивно).
6. Если же после размещения в текущий сервер приложения еще остались (то есть не все приложения размещены), то мы должны перейти к шагу 1, при этом оставив все поместившиеся приложения в текущем сервере. Здесь следует отметить, что переход к шагу 1 в этом случае осуществляется рекурсивным обращением к процедуре, то есть в шаге 1 мы снова начнем цикл с 1 и до количества серверов.
7. Текущий сервер в любом случае освобождается и помечается, как свободный, чтобы его можно было использовать в других вариантах размещения.

- Приложения, которые были в нем размещены, также помечаются, как неиспользованные. Переходим к шагу 1 (не рекурсивно).
8. Если после завершения процедуры стоимость осталась равной максимальной константе, значит, условия для перехода к шагу 4 не были достигнуты и решения задачи для точного распределения нет. В этом случае переходим к шагу 9, иначе – к шагу 16.
 9. Суммируем объемы ресурсов всех приложений, присваиваем это значение переменной w .
 10. Начинаем цикл от 1 до количества серверов.
 11. Если сервер занят, переходим к пункту 10 (не рекурсивно).
 12. Отмечаем текущий сервер как занятый. Если общая производительность серверов больше w , значит, на них можно распределить все приложения, разбив какие-то из них на части. Переходим к шагу 13. Иначе переходим к шагу 14.
 13. Если текущая стоимость всех занятых серверов меньше лучшей на данный момент, то назначаем эту текущую стоимость лучшей.
 14. Переходим к шагу 10 рекурсивно, то есть цикл начинается с 1.
 15. Освобождаем текущий сервер. Переходим к шагу 10 (не рекурсивно).
 16. Если стоимость равна максимальной константе, значит решения нет. Иначе выводим значение стоимости.

На каждом шаге в массивах, отвечающих за занятость серверов и приложений в ячейки, соответствующие занятым приложениям и серверам записывается номер шага, на котором произошло их использование. Благодаря этим данным после работы процедуры появляется возможность восстановить картину размещения приложений на серверах.

Описываются методики оценки производительности аппаратной и программной частей системы для получения данных, необходимых для оптимального распределения ресурсов устройств вычислительной техники, включенных в виртуализованный кластер. Кроме того, приводятся характеристики и даются рекомендации по использованию основных программных средств оценки

производительности. Для реализации описанного выше алгоритма разработана программа RD-calculator.

В ЧЕТВЕРТОЙ ГЛАВЕ приводятся результаты решения трех различных задач планирования ресурсов виртуализованных кластеров, включающих различное число устройств вычислительной техники, развертываемых в ходе миграции, полученные с помощью разработанной в ходе работы над диссертацией программы RD-calculator. Отмечено, что виртуализации подлежат системы, связанные высокоскоростными подключениями (более 10 мбит\сек). Для подключения к виртуализованным сервисам удаленных подразделений предлагается настраивать защищенные каналы через Интернет (так называемые virtual private network (VPN). Термин «расположенные на одной площадке сервера» характеризует одно и\или несколько серверных расположений, соединенных постоянными (возможно резервированными) высокоскоростными подключениями (например, оптическими магистралями).

Представлена серверная инфраструктура трех предприятий, каждая из которых располагается целиком внутри одной площадки и включает в себя определенное количество серверов. Данные о производительности серверов и требованиях к ресурсам приложений получены с помощью инструментария, описанного в 3 главе диссертации. Такая структура является типовой и широко распространена в Российской Федерации и в мире, например, для решения задач автоматизации управления широкого спектра производственных предприятий среднего и большого размера. Программное обеспечение, установленное на серверы таких компаний, включает в себя: различные учетные системы (1С, Парус, Галактика, Microsoft Navision и др.), электронную почту, системы контроля доступа и безопасности, брандмауэры, антивирусные и многие другие. До появления решений виртуализации основным критерием установки и настройки был критерий «одна система – один физический сервер». Отмечается, что в настоящий момент данный критерий практически утратил свое значение.

Произведен анализ ИТ структуры предприятий и даны предложения о степени виртуализации инфраструктуры, затрагивающие виртуализацию разного количества

серверов. Ввод полученных данных в программу RD-calculator и ее исполнение отображает варианты распределения ресурсов виртуализованного кластера. По итогам анализа различных вариантов распределения принимается решение остановиться на варианте, предполагающем оптимальный баланс между производительностью, надежностью и стоимостью решения.

На Рисунке 4 представлен график снижения количества необходимых серверных устройств вычислительной техники после перехода на технологию виртуализации для рассматриваемых задач. Несколько сценариев виртуализации предполагают использование различного количества серверов, критерии использования которых описаны в 4 главе диссертации. Первый столбец графика отображает количество серверов первоначально, второй и третий столбцы – количество серверов после перехода на виртуализованную платформу по двум разным сценариям.

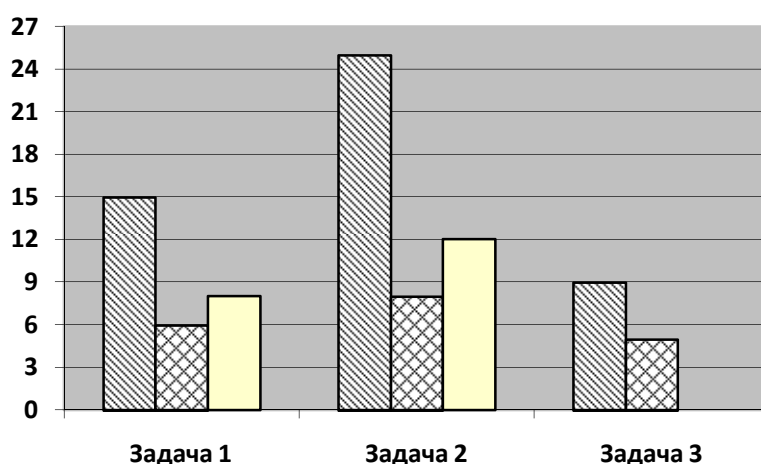


Рисунок 4. Возможность снижения количества серверов после перехода на новую технологию

Средний уровень загрузки серверов по итогам перехода на новую платформу поднялся с 28% до 85% в зависимости от сценария виртуализации.

Снижение количества используемых серверов позволяет минимизировать стоимость обслуживания вычислительной системы в целом. Например, для

экспериментальной задачи № 3 стоимость обслуживания сократилась на 68%, что подтверждает правильность разработанных моделей и алгоритмов.

Таким образом, для рассмотренных задач сделан вывод о технической и финансовой обоснованности перехода на технологии виртуализации, при одновременном повышении характеристик надежности системы в целом.

В ЗАКЛЮЧЕНИИ приводятся основные выводы и результаты выполненной работы.

В ПРИЛОЖЕНИИ приведен листинг программной реализации алгоритма, полученного в ходе работы над диссертацией.

ОСНОВНЫЕ РЕЗУЛЬТАТЫ РАБОТЫ

1. Проведен анализ используемых на практике методов планирования ресурсов виртуализованных кластеров при решении задач управления, выявлено отсутствие целостного универсального подхода.
2. Разработаны математические модели оптимального планирования ресурсов виртуализованного вычислительного кластера, пригодные для применения как для унаследованных аппаратных комплексов, так и для новых систем.
3. Разработан алгоритм и программная реализация предложенных математических моделей, позволяющие получать любое заданное число вариантов распределения ресурсов.
4. Разработана методика применения программного обеспечения для достижения оптимального распределения ресурсов виртуализованного кластера с учетом надежности характеристик элементов и устройств вычислительной техники, включенных в виртуализованный вычислительный кластер.
5. В процессе внедрения разработанного программного обеспечения и предложенной методики проведено экспериментальное исследование на ряде модельных примеров и реальных производственных задач в областях, используемых, в том числе для систем управления.

СПИСОК РАБОТ, ОПУБЛИКОВАННЫХ ПО ТЕМЕ ДИССЕРТАЦИИ

Публикации в изданиях, рекомендованных ВАК РФ

1. В. В. Гольдин, А. Л. Козловский. Оптимизация распределения ресурсов кластеризованной вычислительной автоматизированной системы управления с учетом надежности ее функционирования. // Оборонная техника. Выпуск 6, 7. М., 2011
2. В. В. Гольдин, А. Л. Козловский. Алгоритм решения задачи загрузки виртуализованного кластера. // Оборонная техника. Выпуск 6, 7. М., 2011

В других изданиях

3. В. В. Гольдин, А. Л. Козловский. Математические модели планирования ресурсов вычислительных кластеров. // Материалы научно-практической конференции «Инфо-2010». М., 2010
4. А. Л. Козловский. Эффективный алгоритм решения модифицированной задачи о рюкзаке. // Материалы научно-практической конференции «Инфо-2011». М., 2011
5. А. Л. Козловский. Методы распределения вычислительных ресурсов виртуализованного кластера. // Труды 19-ой Международной научно-технической конференции «Современное телевидение и радиоэлектроника». М., 2011
6. А. Л. Козловский. Программа распределения ресурсов виртуализованного кластера. Свидетельство об официальной регистрации программы для ЭВМ № 2011617318 от 20 сентября 2011 г.