**хостирование приложений в облачной инфраструктуре с применением генетического алгоритма**

***Смирнов Алексей Борисович***

*студент, Череповецкий государственный университет,*

*Россия, г. Череповец*

***Маслов Евгений Александрович***

*научный руководитель, кандидат наук,*

*Череповецкий государственный университет,*

*Россия, г. Череповец*

Оптимальное развертывание приложений на серверах является задачей, требующей внедрения специальных систем балансировки нагрузки на серверах. И если на уровне автоматизированных систем управления предприятием для решения данного вопроса используется такие подходы, как контейнеризация и оркестрация, то на уровне автоматизированных систем управления техническим процессом применяются устоявшиеся технические решения, и программное обеспечение размещается на серверах без применения контейнеризации. Однако, такие ресурсы также нуждаются в управлении, и вручную такие задачи, как, например, оптимизация размещения приложений в серверной инфраструктуре, становится решать крайне трудно.

Данную задачу можно описать следующим образом: «имеется коллекция программных компонентов (далее будем называть их сервисами) и коллекция виртуальных машин (далее будем называть их серверами). Необходимо распределить все сервисы на минимальное количество серверов в предположении, что один сервис может располагаться только на одном сервере». Данная задача может быть описана с использованием терминологии комбинаторной оптимизации, как упаковка множества на , где упаковка — это подсемейство  множеств, такое, что все множества из  попарно не пересекаются, т.е. набор серверов, в которых сервис не входит в два разных сервера, – множество серверов, а - семейство подмножеств серверов. Эта задача, в свою очередь, является классической NP-полной задачей в теории вычислительной сложности и комбинаторике и более известна, как задача об упаковке в контейнеры [1,2].

Неотъемлемой частью алгоритма оптимизации является критерий оптимальности – количественная мера оптимизируемого качества объекта. Значение критерия оптимальности выражается целевой функцией. Таким образом, задача оптимизации сводится к нахождению минимума или максимума целевой функции.

Приведем постановку задачи размещения сервисов на ограниченном количестве серверов. Возьмем - множество сервисов и - множество серверов. Рассмотрим сервисы и сервера с точки зрения задачи упаковки, как предметы и контейнеры, тогда – размер -го предмета (сервиса), а – вместимость го контейнера (сервера). Тогда, требуется найти такое разбиение множества на непересекающиеся подмножества , чтобы сумма размеров сервисов в каждом подмножестве не превосходила заданную вместимость конкретного сервера, и чтобы было наименьшим возможным.

Математическая формулировка данной задачи следующая:

*,*

*,*

где

Можно считать, что сервисы, принадлежащие каждому множеству , размещаются на сервера разного размера, а цель состоит в размещении сервисов из множества на как можно меньшее число серверов .

Критерием оптимальности в данном случае будем считать число серверов . Решение считается тем более оптимальным, чем ниже значение .

Каждый сервис описывается кортежем [3] , где - количество занимаемой оперативной памяти, - количество занимаемой памяти на постоянном запоминающем устройстве, - количество занимаемого процессорного времени, – тип операционной системы, на которой запускается -й сервис. Введем следующие ограничения:

где и – верхние пределы размера файла сервиса, выражаемые в гигабайтах, – верхний предел занимаемого процессорного времени, выраженный в условных единицах, а – элементы множества типов операционных систем.

Каждый сервер описывается кортежем , где - количество свободной оперативной памяти, - общий объем оперативной памяти, - количество свободной памяти на постоянном запоминающем устройстве, – общий объем памяти на постоянном запоминающем устройстве, – общее количество процессорного времени, – тип операционной системы сервера. Введем следующие ограничения:

где и – верхние границы размера файла сервиса, выражаемые в гигабайтах, – верхняя граница занимаемого процессорного времени, выраженный в условных единицах, а – элементы множества типов операционных систем.

В ходе решения задачи положение элементов, обозначенное заданным типом, в пространстве задается множеством .

Для NP-полных вычислительных задач не существует алгоритма решения, способного вернуть результат за полиномиальное время. Решение данной задачи получено путем использования простого алгоритма BFD [5] (наилучший подходящий с упорядочиванием) и генетического алгоритма [4]. Приведено решение для тестовой выборки в 5 серверов и 50 программных компонентов с разными размерами Hdd и Ram. Можно заметить явную тенденцию к сохранению минимального числа занятых серверов при размещении разного количества компонентов на них при использовании генетического алгоритма (таблица 1). Табличные данные представлены в виде графика (рисунок 1).

Таблица 1. Зависимость количества свободных серверов от количества размещаемых программных компонентов

|  |  |  |
| --- | --- | --- |
| Кол-во программных компонентов | Кол-во свободных серверов при применении генетического алгоритма, % | Кол-во свободных серверов при применении алгоритма BFD, % |
| 0 | 100 | 100 |
| 5 | 60 | 60 |
| 10 | 60 | 40 |
| 15 | 60 | 20 |
| 20 | 60 | 0 |
| 25 | 60 | 0 |
| 30 | 40 | 0 |
| 35 | 40 | 0 |
| 40 | 0 | 0 |
| 45 | 0 | 0 |
| 50 | 0 | 0 |

Рисунок 1. График зависимости количества свободных серверов от количества размещаемых сервисов

**Список литературы:**

1. [Смирнов, А.В. О задаче упаковки в контейнеры [текст] / А.В. Смирнов // Общероссийский математический портал. – том 46. – выпуск 4(280). – С. 173-174.](http://www.mathnet.ru/php/archive.phtml?wshow=paper&jrnid=rm&paperid=4649&option_lang=rus)
2. Кормен Томас Х. Алгоритмы: построение и анализ [текст] / Томас Х. Кормен, Чарльз И. Лейзерсон.  — 2-е изд. — М.: [«Вильямс»](https://ru.wikipedia.org/w/index.php?title=%D0%92%D0%B8%D0%BB%D1%8C%D1%8F%D0%BC%D1%81_(%D0%B8%D0%B7%D0%B4%D0%B0%D1%82%D0%B5%D0%BB%D1%8C%D1%81%D1%82%D0%B2%D0%BE)&action=edit&redlink=1), 2006. — 1296 с.
3. Нужнов, Е.В. Трехмерная упаковка несвязных элементов на основе эвристических процедур / Е.В. Нужнов, А.В. Барлит. – Таганрог: Изд-во ТРТУ, 2002. – 23 с.
4. Панченко, Т. В. Генетические алгоритмы: учебно-методическое пособие [Текст] / Т.В. Панченко — Астрахань: Издательский дом «Астраханский университет», 2007. — 87 с.
5. Martello S. Knapsack problems: algorithms and computer implementations // Library of Congress Cataloging-in-Publication Data. – 1990. – P. 221-224.