МОСКОВСКИЙ АВИАЦИОННЫЙ ИНСТИТУТ

(НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ)

Институт №8 «Информационные технологии и прикладная математика»

Кафедра 810 «Информационные технологии в моделировании и управлении»

**Реферат по теме:**

**«Модели распределенных алгоритмов на графах»**

Выполнил: А.С. Бобряков

Группа: М8O-203М-19

Преподаватель: А.И. Майсурадзе

Москва, 2021

**Оглавление**

[**1.** **Введение** 3](#_Toc71555725)

[**2.** **Типы рассматриваемых алгоритмов** 3](#_Toc71555726)

[**3.** **Модели распределенных алгоритмов** 5](#_Toc71555727)

[**4.** **Оценка работы алгоритмов** 6](#_Toc71555728)

[**5.** **Заключение** 7](#_Toc71555729)

[**Список использованных источников** 7](#_Toc71555730)

# **Введение**

В настоящее время при выполнении распределенных алгоритмов над графами возникает потребность в ускорении вычислений путем применения различных вычислительных структур и моделей. Структуру вычислений, на которой решается графовая задача, состоящую из компьютерных кластеров без общей памяти, но с взаимодействием процессоров путем обмена сообщениями, использующих заданную сеть связи, считают совпадающей с графом.

Большинство асинхронных моделей работают на основе одновременного начала работы каждого процессора, расположенного в вершине графа. Но существуют модели, при которых работа начинается с одной корневой вершины. Особенностью таких моделей является, что время работы алгоритма учитывает время начального распространения сообщений от корня до остальных вершин. Это время в синхронном случае равно, так как время доставки сообщения осуществляется за один логический такт (раунд), а в асинхронной не превышает максимального расстояния от корня до других вершин.

В данной работе будут рассмотрены синхронная и асинхронная модели распределенных алгоритмов решения задач на неориентированных графах.

# **Типы рассматриваемых алгоритмов**

Для примера рассматриваются неориентированные нумерованные графы, где каждая вершина хранит свой номер размером, а также переменную множества соседей. На них исполняются алгоритмы решения любых задач под такие графы, разделенных по типам. При том, система основана на сборе информации либо в каждой вершине графа, либо в корне. При решении задачи с разметкой графа происходит соответствующая пометка некоторых вершин или ребер, а ответ о завершении задачи и разметки посылается корню. А в случае решения задачи без разметки корень формирует ответ в виде сообщения вовне.

Первый тип алгоритмов – сбор в корне. Основная структура – рассылается сообщение, которое инициализирует сбор информации о графе, от корня ко всем вершинам. В итоге в корне собирается вся информация о графе, после чего в нем решается основная задача, сообщение с решением которой посылается вовне. Граф при этом не размечается. Цель этого типа - распространение сообщений от корня ко всем ребрам и обратное распространение осуществляются за минимальное время, равное .

Второй тип – сбор сообщений в корне и разметка из корня. Основная структура – сначала выполняется сбор графовой информации в корне, далее в корне решается задача, а затем осуществляется глобальная разметка графа в соответствии с найденным решением. Чтобы осуществить разметку, из корня посылается сообщение во все вершины, чтобы они выполнили свою локальную разметку, после чего оповестили корень о завершении. Когда корень получит информацию о локальном завершении всех вершин, тогда будет завершена глобальная разметка. Цель этого типа - распространение сообщений от корня ко всем ребрам осуществляется за и обратное распространение осуществляется за минимальное время, равное .

Третий тип – сбор и разметка осуществляется во всех вершинах. Основная структура – информация о графе собирается и в корне, и в каждой вершине путем отправки соответствующего сообщения, инициализирующее сбор данных. После сбора информации каждая вершина сама решает поставленную задачу и выполняет локальную разметку. Корень получает соответствующие сообщения о завершении операций на каждой вершине, после чего посылает вовне сообщение о завершении задачи и разметке графа. Цель этого типа – сбор информации в каждой вершине через распространение сообщений от корня ко всем ребрам осуществляется за , и сообщение корню о завершении разметки осуществляется за .

# **Модели распределенных алгоритмов**

Модель определяется структурой, при которой процессоры выполнения задач находятся в вершинах графа и обмениваются между собой сообщениями, которые пересылаются по рёбрам. Граф предполагается неориентированным, связным, упорядоченным без кратных рёбер и петель, с выделенной вершиной, являющейся корнем графа. Ребра являются двунаправленным каналом передачи сообщений.

В отличие от стандартных подходов выполнения задач рассматриваемые модели запускают задачу не одновременно на всех узлах, а только с корня. Другие вершины начинают выполнение задачи после получения соответствующего сообщения, передаваемого по вершинам. Соответственно, задача считается решённой только после того, как корень получил об этом информацию.

Рассматривается две разновидности модели: синхронная – RSM (Rooted Synchronous Model) и асинхронная – RAM (Rooted Asynchronous Model). В синхронной модели сообщение доставляется по ребру за фиксированное время. В асинхронной модели сообщение доставляется за произвольное время, ограниченное сверху 1 тактом. Но время передачи сообщения по ребру может быть различным в зависимости от самих рёбер, направлений, и также оно меняеться со временем. В RSM между раундами проходит ровно один такт, и время работы алгоритма можно считать, как число раундов. Под раундом понимается каждая итерация, при которой процессоры в вершинах принимают все сообщения, обрабатывают их и посылают новые. В RAM время между раундами находится в интервале (0,1], поэтому нет смысла считать время работы алгоритма в раундах. Нужно оценивать время в тактах при наихудшем случае в зависимости от времени передачи по рёбрам сообщений во все моменты времени.

# **Оценка работы алгоритмов**

Оценка работы алгоритмов приведена в таблице 1. Под понимается время работы алгоритма, – размер памяти вершины (сумма размеров хранимых переменных без учета памяти для принимаемы и посылаемых сообщений), – максимальное расстояние от корня до вершины, – диаметр графа, – размер описания графа, необходимого для решения задачи, – число вершин, Δ - максимальная степень вершины, - максимальный суммарный размер сообщений, находящихся на всех рёбрах одновременно.

Таблица 1 – Оценка работы алгоритмов по их типам.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Тип 1 | | | Тип 2 | | | Тип 3 | |
| RSM | | RAM | RSM | | RAM | RSM | RAM |
| (для некорневой вершины) | *O(min{n,Δlogn})* | | *O(n)* | *O(Δlogn) = O(nlogn)* | | *O(n+ Δlogn) = O(nlogn)* | *O(P+n+ Δlogn) = O(n2)* | |
| (для корневой вершины) | *O(P +logn) = O(n2)* | | *O(P + +n) = O(n2)* | *O(P + Δlogn) = O(n2)* | | *O(P + +n+ Δlogn) = O(n2)* | *O(P+n+ Δlogn) = O(n2)* | |
|  | *≤ 2+1* | | | *≤ 4+1* | | | *≤ 2+ +1* | |
|  | O(n2logn) | O(n4) | | O(n3) | O(n4) | | O(n4) | |

# **Заключение**

В работе приводится обзор синхронной и асинхронной распределенных моделей для решений нескольких типов задач на основе нумерованных неориентированных графов, а также приводятся оценки их сложности. Рассматриваемые набор задач содержит задачи, основанные на сборе информации о всем графе в корне или в вершине, а также задачи, проводящие разметку графа. Для этого набора приведена характеристика памяти, отвечающая за максимальный суммарный размер сообщений, находящихся на всех рёбрах одновременно.

# **Список использованных источников**

1. И. Бурдонов, А. Косачев. Распределённые алгоритмы на корневых неориентированных графах // Труды Института системного программирования РАН. Том 29, выпуск 5, 2017, стр. 283-310.
2. И. Бурдонов, А. Косачев. Общий подход к решению задач на графах коллективом автоматов. Труды ИСП РАН, том 29, вып. 2, 2017, стр. 27-76.
3. И. Бурдонов, А. Косачев. Проблема отката в ориентированной распределенной системе // Труды Института системного программирования РАН. Том 30, выпуск 2, 2018, стр. 167-194.