

РАЗМЕЩЕНИЕ РЕПЛИК МАССИВОВ ДАННЫХ В ДИНАМИЧНЫХ РАСПРЕДЕЛЕННЫХ СИСТЕМАХ

Сомов С.К.

*Институт проблем управления им. В.А. Трапезникова РАН,
Россия, г. Москва, ул. Профсоюзная д.65
ssomov2016@ipu.ru*

Аннотация: В работе представлен метод адаптивного перераспределения массивов данных и их реплик среди узлов распределенной системы, значения параметров которой динамичны и могут изменяться со временем. Метод предназначен для адаптации размещения данных, используемых распределенной системой, к изменившимся условиям ее функционирования путем выбора наилучшего варианта перераспределения массивов данных и их реплик между узлами сети. Цель предлагаемого метода заключается в поиске нового распределения массивов данных и их реплик, обеспечивающего минимум затрат на эксплуатацию системы с учетом затрат на миграцию реплик.

Ключевые слова: распределенные системы обработки данных, перераспределение массивов данных и их реплик, миграция реплик массивов.

Введение

Для повышения эффективности работы распределенных систем обработки данных (РСОД) используются разнообразные методы. Одним из часто используемых методов является использование в системе информационной избыточности, согласно которому в дополнение к массивам данных в узлах сети размещаются и их реплики [1-3].

В течение последних десятилетий было опубликовано много работ, посвященных вопросам оптимального размещения в узлах распределенных систем массивов данных (файлов данных, таблиц и фрагментов таблиц данных распределенных баз данных) и их реплик [3-6].

В основном в данных работах рассматривается задача определения оптимального размещения массивов и их реплик в распределенной системе при условии, что параметры самой системы и внешние условия ее функционирования статичны и не меняются на длительном отрезке времени работы системы. На практике, однако, часто бывает так, что изменяются параметры самой системы (топология сети, характеристики каналов связи и узлов сети) и условия ее функционирования (частота запросов, генерируемых в узлах сети, появление новых запросов и перечня данных, необходимых для их обработки). Это может привести к снижению эффективности работы распределенной системы.

В данной работе представлен метод перераспределения массивов данных и их идентичных реплик между узлами распределенной системы, предназначенный для адаптации системы к изменившимся параметрам ее функционирования. Изменения параметров могут быть настолько значительными, что могут привести к существенному снижению эффективности работы системы, что обуславливают необходимость адаптации системы к новым условиям работы.

Предлагаемый метод предназначен для адаптации распределенной системы к новым условиям ее функционирования путем поиска оптимального перераспределения массивов данных и их реплик между узлами сети. Оптимальность варианта перераспределения определяется на основе двух критериев: минимума стоимостных затрат на эксплуатацию системы или минимума среднего времени реакции системы на запросы к данным.

При подсчете стоимостных и временных характеристик распределенной системы учитываются следующие особенности обработки запросов к массивам данных или их репликам [2,3,6]:

- Информационные запросы адресуются для обработки только в один узел с необходимой репликой данных, находящийся на кратчайшем пути от узла-источника запроса.
- Запросы на модификацию данных одновременно адресуются из узла-источника по кратчайшим путям во все узлы сети, в которых размещены реплики массива и обрабатываются синхронно в рамках одной распределенной транзакции [5-7]. При этом считается, что такой запрос успешно обработан, если получены ответы об успешном обновлении реплик во всех N узлах. Время обработки запроса на модификацию данных будет равно максимальному времени модификации данных в одном из всех N узлов с репликами модифицируемого массива данных.

1 Формальная модель распределенной системы, использующей реплики массивов данных

Предполагается, что РСОД функционирует на основе компьютерной сети, состоящей из K узлов: $N = \{N_1, \dots, N_k, \dots, N_K\}$. Топология сети представлена взвешенным графом $G = (K, \Gamma)$.

В распределенной системе используется множество A из M различных массивов данных $A = \{A_1, \dots, A_m, \dots, A_M\}$, каждый из которых может иметь некоторое количество идентичных реплик, размещенных в узлах сети.

Задана матрица $Dt = \{dt_{ij}\}$ смежности взвешенного графа G , элемент dt_{ij} которой это среднее время передачи единицы данных между парой смежных узлов сети N_i, N_j ($i, j = \overline{1, K}$). Величина элемента dt_{ij} рассматривается как вес дуги графа между парой смежных узлов (i, j).

На основе матрицы Dt с использованием любого известного алгоритма поиска кратчайших путей в графе (например, алгоритм Дейкстры [8]), строится матрица $DTC = \{DTC_{ij}\}$, $i, j = \overline{1, K}$ средних времен передачи единицы данных по кратчайшим путям между узлами сети.

Определена матрица $CUC = \{CUC(i, j)\}$, элемент которой $CUC(i, j)$ это стоимость передачи сообщения единичной длины по каналу связи между парой смежных узлов (i, j).

На основе этой матрицы рассчитываются матрица SPC затрат на передачу сообщений единичной длины по кратчайшим путям между узлами сети.

Для каждого узла N_k сети определена максимальная емкость C_k устройств хранения данных узла и максимальное количество AM_k реплик различных массивов данных, которые могут быть размещены в этом узле.

Для каждого массива данных и его реплик задан его размер (объем) $AV(A_m)$.

2 Параметры задачи размещения реплик массивов данных

Запросы пользователей системы к данным обрабатывается множеством $p \in \{1, \dots, P\}$ прикладных процессов, исполняемых в узлах сети. В одном узле может выполняться несколько различных процессов. Каждый процесс во время своей работы генерирует некоторое количество информационных запросов и/или запросов на модификацию данных одного или нескольких массивов данных. Характеристики прикладных процессов заданы в следующих матрицах:

- Частота выполнения прикладных процессов в узлах сети задается матрицей $PF = \|PF_{kp}\|$ ($k = \overline{1, K}$; $p = \overline{1, P}$). Элемент pf_{kp} матрицы это частота решения p -го процесса в k -м узле сети.
- Частота генерации p -м прикладным процессом информационных запросов к массиву A_m определяется матрицей $PQE = \|PQE_{pm}\|$, где элемент PQE_{pm} это частота генерации p -м процессом информационных запросов к массиву данных A_m .
- Частота генерации p -м прикладным процессом запросов на модификацию данных массива A_m определяется в матрице $PQU = \|PQU_{pm}\|$, элемент PQU_{pm} которой равен частоте генерации p -м процессом запросов на модификацию данных массива A_m .

В формальной модели распределенной системы предполагается, что:

- В узлах сети бесконечные очереди на обслуживание запросов и распределенная система работает в установившемся режиме, без сбоев. Тогда все запросы к данным обрабатываются распределенной системой успешно.
- Все запросы к данным реплики конкретного массива данных обрабатываются прикладными процессами в том же узле сети, в котором размещена эта реплика.

Определены следующие характеристики распределенной системы:

- кортеж $T^E = (T_1^E, \dots, T_k^E, \dots, T_K^E)$, элемент T_k^E которого это среднее время обработки информационных запросов в узле k сети,
- кортеж $S^E = (S_1^E, \dots, S_k^E, \dots, S_K^E)$, в котором S_k^E это средняя стоимость использования ресурсов узла k сети за единицу времени обработки информационных запросов в этом узле,
- кортеж $T^U = (T_1^U, \dots, T_k^U, \dots, T_K^U)$ со средними временами обработки в различных узлах сети запросов на модификацию данных реплик,

- кортеж $S^U = (S_1^U, \dots, S_k^U, \dots, S_K^U)$, элемент S_k^U которого это средняя стоимость использования ресурсов узла k сети за единицу времени обработки запросов на модификацию данных,
- $Evol$ – средний размер (объем) информационного запроса,
- $Erepvol$ – средний размер ответа на информационный запрос,
- $Uvol$ – средний размер (объем) запроса на модификацию данных,
- $Urepvol$ – средний размер ответа на запрос на модификацию данных,
- кортеж $AV(A_m)$ с размерами массивов данных и их реплик,
- кортеж $Dsc = (Dsc_1, \dots, Dsc_k, \dots, Dsc_K)$ со стоимостью хранения единицы данных в узлах сети,
- матрица RSC , элемент $RSC(N_k, A_m)$ которой это затраты на хранение массива A_m или его реплики в узле N_k .

Размещение массивов данных и их реплик в узлах сети задается матрицей $X = \{x_{mk}\}$, в которой: $x_{mk} = 1$, если реплика массива A_m размещена в узле k , и $x_{mk} = 0$ в противном случае.

3 Величина затрат на функционирование системы

Величина затрат на функционирование распределенной системы, использующей реплики массивов данных, состоит из нескольких компонент.

3.1 Затраты на хранение в узлах системы массивов данных и их реплик

Затраты $TRSC$ на хранение массивов данных и их реплик определяются с использованием данных матрицы X распределения реплик и матрицы стоимости хранения реплик в узлах сети $RSC(N_k, A_m)$:

$$TRSC = \sum_{m=1}^M \sum_{k=1}^K x_{mk} \times RSC(k, m)$$

3.2 Затраты на использование каналов связи

Затраты на передачу по каналам связи всех запросов и ответов на эти запросы, генерируемые во всех узлах системы за единицу времени, будут равны $CRRT = CRRT_E + CRRT_U$. Здесь:

$CRRT_E$ – затраты на передачу информационных запросов и ответов на них, сгенерированных во всех узлах сети за единицу времени.

$$CRRT_E = \sum_{k=1}^K \sum_{p=1}^P PF(k, p) \sum_{m=1}^M [PQE(p, m) * SPC(k, w) * x_{mk} * (Evol + Erepvol)]$$

Здесь $SPC(k, w)$ – стоимость передачи сообщения единичной длины из узла N_k по кратчайшему пути в узел N_w с репликой массива данных m , необходимой для обработки запроса. Т.е.:

$$SPC(k, w) = \min_{j=1, \dots, K} SPC(k, j), \quad x_{mj} \neq 0$$

$CRRT_U$ – затраты на передачу запросов (и ответов на них) на модификацию массивов данных, сгенерированных в системе за единицу времени.

$$CRRT_U = \sum_{k=1}^K \sum_{p=1}^P PF(k, p) \sum_{m=1}^M PQU(p, m) * \sum_{w=1}^K x_{mw} * SPC(k, w) * (Uvol + Urepvol)$$

3.3 Затраты на обработку запросов в узлах сети с репликами массивов данных

Величина RPC затрат на обработку всех запросов, сгенерированных в системе за единицу времени, равна сумме: $RPC = RPC_E + RPC_U$, в которой:

RPC_E – затраты на обработку информационных запросов.

$$RPC_E = \sum_{k=1}^K \sum_{p=1}^P PF(k, p) \sum_{m=1}^M PQE(p, m) * \sum_{w=1}^K x_{mw} * S^E(w, m)$$

RPC_U – затраты на обработку запросов на модификацию массивов данных.

$$RPC_U = \sum_{k=1}^K \sum_{p=1}^P PF(k, p) \sum_{m=1}^M PQU(p, m) * \sum_{w=1}^K x_{mw} * S^U(w, m)$$

В этих двух формулах узел w это узел сети, в котором размещена реплика массива A_m , и путь до этого узла от узла N_k является кратчайшим, т.е.:

$$DTC(k, w) = \min_{j=1, \dots, K} DTC(j, w), \quad x_{mj} \neq 0$$

3.4 Затраты на миграцию реплик массивов данных между узлами сети

После того, как определено новое распределение X реплик массивов данных выполняется перемещение (миграция) реплик в новые места их дислокации (другие узлы сети).

Процесс миграции реплик использует данные двух матриц: X – новое распределение реплик и X^1 – предыдущее распределение реплик.

Затраты RMC на миграцию реплик равны:

$$RMC = \sum_{m=1}^M \sum_{k=1}^K x_{mk} [SPC(n, k) * AV(A_m)]$$

Здесь n это номер узла сети, в котором размещена реплика массива m ($x_{mn}^1 = 1$), и который находится на кратчайшем пути от узла k . В узел k реплика массива передается из узла n по кратчайшему пути между этими узлами.

Таким образом, общие затраты SOPC на функционирование системы эта сумма следующих компонент:

$$SOPC = TRSC + CRRT + RPC + RMC$$

4 Среднее время обработки запросов в системе

В предложенной модели предполагается, что система работает в установившемся режиме, а очереди запросов на обслуживание бесконечны и, следовательно, все запросы обрабатываются. Тогда среднее время $AvRPT$ обработки запроса в системе рассчитывается в результате деления общего времени $ToRPT$ обработки в системе всех запросов за единицу времени на общее количество $ToNR$ этих запросов:

$$AvRPT = ToRPT / ToNR$$

В этой формуле:

$$ToRPT = \sum_{k=1}^K \sum_{p=1}^P PF(k, p) \left(\sum_{m=1}^M PQE(p, m) \sum_{j=1}^K x_{mj} T^E(j, m) + \sum_{m=1}^M PQU(p, m) \sum_{j=1}^K x_{mj} T^U(j, m) \right)$$

В формуле для расчета $ToRPT$ узел j это такой узел сети, в котором размещена реплика массива A_m , и путь до этого узла от узла N_k , источника запроса, является кратчайшим, т.е.:

$$DTC(k, j) = \min_{w=1, \dots, K} DTC(k, w), \quad x_{mw} \neq 0$$

$$ToNR = \sum_{k=1}^K \sum_{p=1}^P PF(k, p) \times \sum_{m=1}^M [PQE(p, m) + PQU(p, m) * PQU_r(m)]$$

Здесь PQU_r это количество реплик массива данных m , размещенных в узлах сети, включая и сам оригинал массива, которое равно:

$$PQU_r(m) = \sum_{k=1}^K x_{mk}$$

5 Приоритет узлов сети, генерирующих запросы к массивам данных и репликам

В предлагаемой модели используется параметр $NP_{km} = NP(N_k, A_m)$ - «приоритет» узла сети N_k , в котором генерируются запросы к массиву данных A_m . Значение приоритета NP_{km} равно количеству всех запросов, генерируемых за единичный интервал времени в узле N_k к данным массива A_m :

$$NP_{km} = \sum_{p=1}^P PF(k, p) * [(PQE(p, m) + PQU(p, m))]$$

В процессе размещения в узлах сети реплик каждого массива m , первый экземпляр реплики размещается в узле k с наибольшим значением приоритета NP_{km} , второй экземпляр реплики – в узле

с таким же или меньшим значением приоритета и т.д. Такой процесс продолжается до того момента, когда в сети останутся узлы, приоритет которых меньше заданного минимального значения NP_{min} . В этих узлах реплики не размещаются.

6 Формулировка задачи оптимального размещения реплик массивов данных

Необходимо найти такое распределение X реплик массивов данных по узлам распределенной системы, которое обеспечивает минимальную величину SOPC затрат на функционирование системы или минимальное среднее время AvRPT ответа на запрос к репликам массивов данных:

$$\min_X \text{SOPC или } \min_X \text{AvRPT}$$

Распределение реплик массивов данных должно удовлетворять следующим ограничениям.

- Общий объем реплик размещенных в одном узле сети, не должен превышать емкость узла:

$$\sum_{m=1}^M x_{mk} * Vol(A_m) \leq C_k, \quad 1 \leq k \leq K$$

- В каждом узле сети количество реплик не должно превышать заданного лимита:

$$\sum_{m=1}^M x_{mk} \leq AM_k, \quad 1 \leq k \leq K$$

- Оригинал каждого массива данных A_m должен быть распределен в одном из узлов сети:

$$\sum_{k=1}^K x_{mk} = 1, \quad 1 \leq m \leq M$$

В рассматриваемой модели предполагается, что параметры распределенной системы динамичны и могут менять свои значения с течением времени. Например, со временем может измениться трафик запросов в узлах сети или изменится топология сети. В такой ситуации возникает необходимость перераспределения реплик по узлам сети, т.е. необходимо решить сформулированную выше задачу поиска нового оптимального распределения реплик.

Большая вычислительная сложность задач такого типа предполагает использование эвристических алгоритмов из решения. Ниже представлена блок-схема рассматриваемого алгоритма.

7 Описание эвристического алгоритма размещения реплик массивов данных

Эвристический алгоритм размещения реплик массивов данных в узлах распределенной системы состоит из следующих шагов:

Шаг 1. Вычисление приоритетов узлов и заполнение матрицы NP . Определение диапазона изменения значений ограничения $NPmin$. Вычисление $NPdlt$ шага изменения ограничения на приоритеты узлов.

Шаг 2. Цикл поиска распределения массивов с очередным значением ограничения $NPmin$:

$NPmin = NPmin + NPdlt$.

Если $NPmin$ превышает максимальное значение $NPmax$, то алгоритм заканчивает работу.

Шаг 3. Первоначальное распределение реплик всех массивов данных по узлам с наибольшими приоритетами. Один массив – одна реплика.

Шаг 4. Размещение реплик массивов в порядке убывания приоритета узлов.

Проверка того, что приоритет узла больше минимального значения.

Шаг 5. Миграция реплик. Удаление невостребованных реплик. Выполняется процесс миграции реплик массивов в соответствии с новым распределением X и распределением v_1 , полученным при предыдущем решении задачи. Затем удаляются невостребованные реплики из тех узлов, величина приоритета для которых меньше значения ограничения $NPmin$.

Шаг 6. Проверка ограничений задачи.

Если ограничения не выполнены, то найденное распределение X реплик не оптимально и происходит возврат к шагу 2.

Иначе запоминаем текущее распределение X как оптимальное. Запоминаем текущее значение $NPmin$. Возврат к шагу 2.

Шаг 7. Окончание работы алгоритма.

Найдено оптимальное распределение X .

Ниже на рисунке 1 представлена блок-схема предложенного эвристического алгоритма размещения реплик массивов данных, используемых в распределенной системе с динамично меняющимися параметрами.

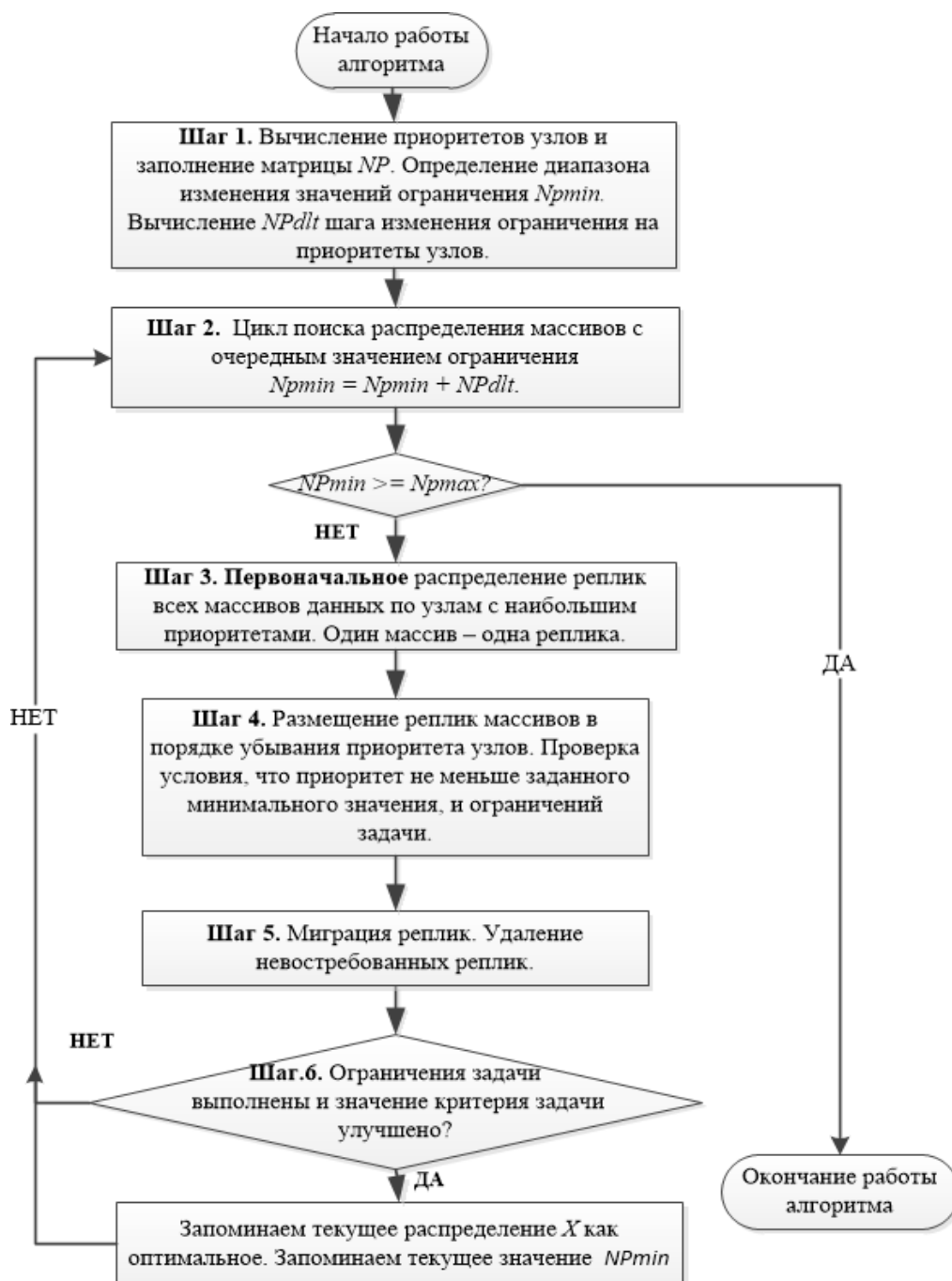


Рис.1. Блок-схема алгоритма

8 Пример работы эвристического алгоритма размещения реплик

Рассматривается распределенная система, имеющая параметры: $K = 10$ – число узлов в сети, $P = 6$ – количество прикладных процессов обработки запросов, $M = 6$ – количество различных массивов данных. Каждый массив может иметь несколько реплик, размещенных в разных узлах сети. Топология сети представлена взвешенным графом $G = (K, \Gamma)$. Граф показан на рисунке 2. Дуги графа раскрашены средним временем dt_{ij} передачи единицы данных между парами смежных узлов графа.

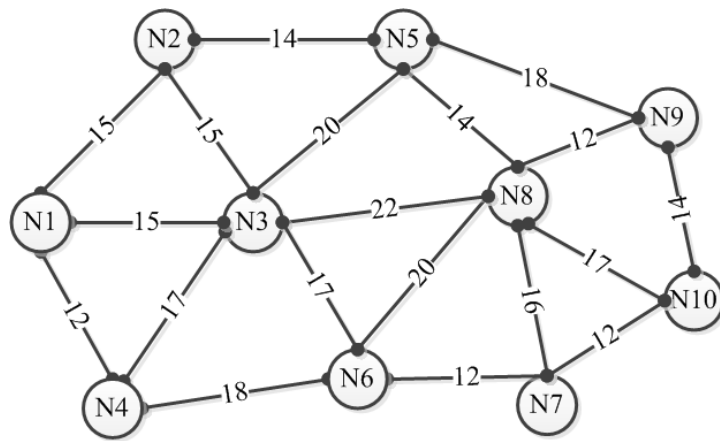


Рис. 2. Топология компьютерной сети

8.1 Работа алгоритма при первоначальных параметрах работы системы

В нескольких таблицах ниже представлены параметры распределенной системы, используемые эвристическим алгоритме при поиске оптимального распределения реплик.

Таблица 1. Значения элементов кортежей параметров

Кортеж параметров системы	Узлы сети									
	1	2	3	4	5	6	7	8	9	10
T_e - среднее время обработки информационных запросов	30	50	40	50	40	30	20	50	40	20
T_u - среднее время обработки запросов на модификацию данных	61	81	72	83	74	85	56	79	09	49
S_e - стоимость использования ресурсов узлов сети при обработке информационных запросов	26	36	35	47	38	38	49	49	56	66
S_u - стоимость использования ресурсов узлов сети при обработке запросов на модификацию данных	055	46	67	38	46	55	77	275	368	476
D_{sc} - стоимость хранения единицы данных в узлах сети	10	20	15	22	20	10	14	15	12	14

Таблица 2. Частоты PF выполнения прикладных процессов в узлах сети

Узел сети	Процессы					
	1	2	3	4	5	6
1	3	4	4	12	1	2
2	2	2	13	2	2	1
3	11	2	1	1	9	2
4	2	3	1	4	23	8
5	4	2	2	2	5	12
6	5	3	2	1	14	1
7	3	3	10	3	4	4
8	2	5	8	5	7	5
9	1	1	5	2	12	24
10	7	3	6	3	8	7

Таблица 3. Частоты PQE и PQU генерации процессами запросов к данным

PQE - Частоты информационных запросов							PQU - Частоты запросов на модификацию данных						
Процесс	Массивы данных						Процесс	Массивы данных					
	1	2	3	4	5	6		1	2	3	4	5	6
1	9	3	26	4	7	4	1	1	2	48	13	3	5
2	1	2	3	14	2	2	2	3	2	3	24	4	4
3	2	3	3	2	23	3	3	3	4	4	2	19	2
4	3	1	2	1	2	23	4	3	5	4	2	5	22
5	2	25	3	2	1	2	5	9	24	4	4	2	5
6	25	2	2	3	4	2	6	19	4	3	3	13	7

В процессе работы алгоритма на основе исходных данных рассчитываются элементы матрицы NP приоритетов узлов сети. Матрица представлена в таблице 4.

В результате работы алгоритма найдено оптимальное распределение X при ограничении на минимальное значение приоритета $NP_{min} = 534$ и количестве массивов данных и их реплик, размещенных в сети, равном 14 (6 массивов данных и 8 реплик этих массивов). При этом получено оптимальное значение затрат на функционирование системы, равное 359,196,032.00 условных единиц стоимости. Полученному оптимальному решению соответствует матрица X, показанная в таблице 5, и граф на рисунке 3, иллюстрирующий оптимальное распределение реплик массивов данных.

Таблица 4. Матрица NP приоритетов узлов.

Узел сети	Массивы данных					
	1	2	3	4	5	6
1	237	192	363	273	343	636
2	171	225	282	186	615	208
3	316	529	912	336	232	242
4	666	1228	398	350	313	454
5	653	371	429	260	369	291
6	276	749	511	300	218	225
7	330	335	376	262	569	294
8	407	489	338	343	527	407
9	1239	788	331	297	684	430
10	526	541	687	356	504	365

Таблица 5. Распределение X реплик по узлам сети

Узел сети	Размещение реплик массивов данных					
	1	2	3	4	5	6
1	0	0	0	0	0	1
2	0	0	0	0	1	0
3	0	1	1	0	0	0
4	1	1	0	0	0	1
5	1	0	0	0	0	0
6	0	1	1	0	0	0
7	0	0	0	0	1	0
8	0	1	0	0	1	0
9	1	1	0	0	1	0
10	1	1	1	1	1	0

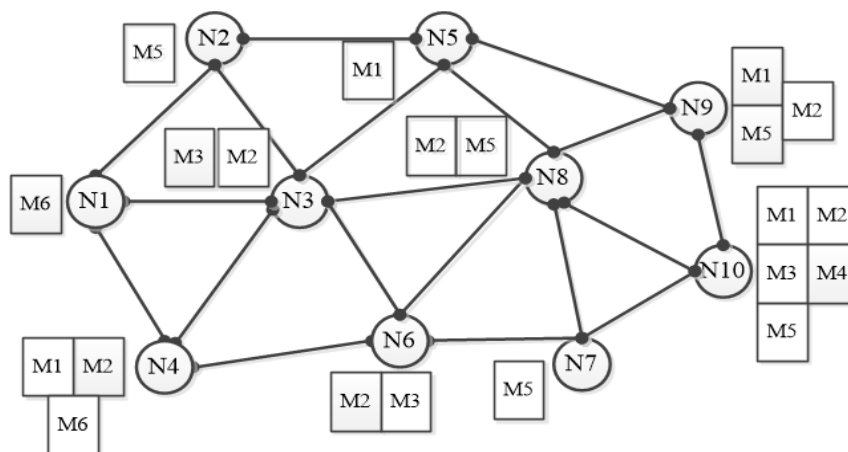


Рис. 3. Распределение реплик массивов по узлам сети

8.2 Работа алгоритма при измененных параметрах работы системы

Предположим, что по истечении некоторого времени работы распределенной системы произошли существенные изменения в частотах выполнения процессов в узлах распределенной системы. Измененные значения частот процессов представлены в таблице 6.

Таблица 6. Новые частоты PF выполнения прикладных процессов в узлах сети

Узел сети	Процессы					
	1	2	3	4	5	6
1	3	4	6	12	8	12
2	2	2	13	2	2	1
3	11	2	1	1	9	2
4	2	3	1	4	23	8
5	4	2	2	2	5	12
6	5	3	2	1	14	1
7	12	3	3	3	4	4
8	2	5	8	5	7	5
9	1	1	5	2	12	24
10	3	3	2	3	1	1

Изменение частот выполнения процессов в узлах сети привело к пересчету приоритетов узлов. Обновленная таблица NP приоритетов узлов представлена ниже таблица 7. Изменились значения в следующих строках матрицы NP : 1,7,8,10.

Изменения значений приоритетов узлов вызвало необходимость перераспределения реплик массивов данных для того, чтобы наилучшим образом адаптировать работу распределенной системы к новым условиям функционирования.

Матрица X с описанием оптимального распределения, представлена в таблице 8.

На рисунке 4 показана сеть с новым оптимальным размещением реплик массивов данных.

Таблица 7. Приоритеты узлов сети

Узел сети	Реплики массивов данных					
	1	2	3	4	5	6
1	764	609	476	383	618	785
2	171	225	282	186	615	208
3	316	529	912	336	232	242
4	666	1228	398	350	313	454
5	653	371	429	260	369	291
6	276	749	511	300	218	225
7	385	331	993	387	365	340
8	1005	682	1142	605	1228	763
9	1239	788	331	297	684	430
10	125	114	284	194	173	206

Таблица 8. Новое распределение реплик

Узел сети	Оптимальное размещение массивов данных и их реплик					
	1	2	3	4	5	6
1	1	0	0	0	0	1
2	0	0	0	0	0	0
3	0	0	1	0	0	0
4	0	1	0	0	0	0
5	0	0	0	0	0	0
6	0	1	0	0	0	0
7	0	0	1	0	0	0
8	1	0	1	1	1	1
9	1	1	0	0	0	0
10	0	0	0	0	0	0

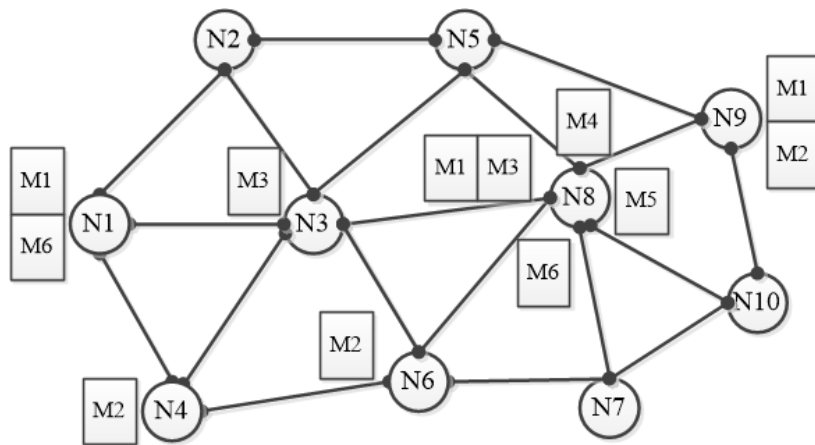


Рис. 4. Новое распределение реплик массивов по узлам сети

В итоге работы алгоритма с измененными значениями параметров системы получено оптимальное распределение реплик с наименьшими затратами 344,445,103.00 на эксплуатацию системы при значении минимального приоритета узлов, равного 744 и при распределении узлах сети 6 массивов данных и 7 их реплик. В результате перераспределения реплик получено уменьшение затрат на функционирование системы с суммы 359,196,032.00 до 344,445,103.00 условных единиц стоимости.

Заключение

В работе представлена формальная модель динамичной распределенной системы, использующей для своей работы массивы данных и их реплики. Динамичность распределенной системы заключается в том, что некоторые параметры распределенной системы могут быть изменчивы и со временем меняют свои значения. В рамках этой модели сформулирована задача поиска оптимального перераспределения реплик с тем, что бы достиглось минимально возможные затраты на функционирование распределенной системы или минимальное среднее время обработки запросов. Учитываются затраты на хранение массивов данных и их реплик, затраты на передачу запросов по каналам связи и затраты на обработку запросов в узлах сети. В силу большой вычислительной сложности сформулированной задачи для ее решения предложен эвристический алгоритм, работа которого проиллюстрирована на примерах.

Предложенный эвристический алгоритм реализован на языке C++ в среде MS Visual Studio 2019. Работа выполнена в рамках фундаментального исследования № 0052-2019-0011 ИПУ РАН

Литература

1. Микрин Е.А., Сомов С.К. Обзор моделей и методов обеспечения сохранности данных в распределенных системах обработки данных // Информационные технологии и вычислительные системы. – 2017. – № 4. – С. 5–28.
2. Сомов С.К. Сохранность информации в распределенных системах обработки данных. Научное издание // – М.:ИПУ РАН. 2019. – 254с.
3. Чернышев Г.А. Обзор подходов к организации физического уровня в СУБД // Труды СПИИРАН. 2013. – Санкт-Петербург, 2013. Вып. 1(24). – С. 222 – 275.
4. Tambulea L., Horvat. M. Dynamic Distribution Model in Distributed Database // Int. J. of Computers, Communications & Control, Vol. 3. 2008. – P. 512-515.
5. Abdalla H.I. A New Data Re-Allocation Model for Distributed Database Systems // International Journal of Database Theory and Application. Vol.5. 2012. №.2, – P.45-60.
6. Ozsu M. T., Valduriez P. Principles of Distributed Database Systems, 4th Edition // Springer. 2020. — P. 674.
7. Таненбаум Э., М. ван Стеен. Распределенные системы. Принципы и парадигмы // — СПб.: Питер. — 2003. — 877 с.
8. Майника Э. Алгоритмы оптимизации на сетях и графах: Перевод с английского // — М.: Мир. — 1981. — 323 с.