

ДЛИТЕЛЬНОСТЬ МИГРАЦИИ ВИРТУАЛЬНЫХ СЕРВЕРОВ В РАСПРЕДЕЛЕННОЙ СИСТЕМЕ

В статье рассмотрена задача о длительности миграции виртуального сервера в распределенной системе. Представлено решение этой задачи в рамках двух математических моделей. Модели реализованы программно. Результаты продемонстрированы на конкретном примере.

Ключевые слова: распределенная система, миграция, виртуальный сервер, N - k схема хранения файлов, поиск частей файла, имитационное моделирование, эмпирическая функция распределения, вероятностное пространство.

Задача о миграции

Базовые термины. Под *распределенной системой* будем понимать множество компьютеров, соединенных в сеть. На каждой машине системы могут быть запущены несколько *виртуальных серверов*, т. е. наборов программ, имитирующих реальный компьютер. Виртуальные серверы выполняют различные (обычно сетевые) задачи. Для хранения данных эти задачи используют виртуальные диски, представляющие собой специальные файлы крупного объема. Таким образом, дисковые данные виртуального сервера имеют четко определенный размер.

Нередко требуется *смигрировать* виртуальный сервер с одной машины распределенной системы на другую, т. е. скопировать его данные, необходимые для запуска. Такая потребность возникает, когда требуется освободить компьютер для обновления ее программного или аппаратного обеспечения. Для виртуального сервера, выполняющего сетевые задачи, может потребоваться миграция на машину, более «близкую» (в сетевом смысле) к клиентам для улучшения доступа к нему. Подробнее о виртуальных серверах и их миграции см. в [Pinheiro, 2002].

Актуальность задачи. Сервер – это роль компьютера в сети. Виртуальный сервер имитирует реальный; т. е. обрабатывает сетевые запросы от клиентов. Одной из самых существенных характеристик сервера является его доступность. В любой момент времени он должен быть готовым обработать поступивший запрос. Длительность миграции виртуального сервера – это то время, в течение которого он будет недоступным для клиентов. Поэтому крайне важно знать оценку этой величины. Возможно, миграцию вообще стоит отменить, если длительность окажется слишком долгой.

Еще большую ценность представляет собой способ определения зависимости длительности миграции от характеристик распределенной системы. Он позволит выделить степень влияния на длительность отдельных факторов, а также насколько сократится время миграции при их улучшении.

Схема хранения файлов. Файлы в распределенной системе хранятся особым образом: каждый файл разбивается на блоки стандартного размера. Каждый блок разбивается на порции по так называемой N - k схеме. N и k – параметры этой схемы. Блок разбивается на N порций, причем так, что *из любых k порций можно собрать весь блок целиком!* (N всегда больше или равно k).

Таким образом, один файл хранится в системе, хоть это и несколько избыточно, для того, чтобы повысить надежность и скорость сборки файла – процедуры сбора каждого блока файла из порций и самого файла из этих блоков. Подробнее об N - k схеме см. в [Хасин, 2001; Castro et al., 2004].

Процедура сборки файла. Сборка файла производится в предположении полной неопределенности расположения его порций. Пусть процедура сборки запущена на некотором цен-

тральном узле. Чтобы собрать блок файла, необходимо в первую очередь получить информацию о местонахождении необходимых для этого порций. Для этого центральный узел рассылает соответствующий запрос всем своим соседям. Любой узел, получив впервые такой запрос:

- если он содержит требуемую порцию, отправляет обратно сообщение об этом;
- рассылает запрос всем своим соседям, кроме общих с узлом, пославшим ему запрос.

Каждый запрос имеет характеристику TTL (time-to-live) – максимальное количество узлов, через которые он может пройти. Пропуская через себя запрос, любой узел уменьшает это количество на единицу. Когда же оно становится равным нулю, запрос уничтожается. Сообщение об этом достигает центрального узла по обратному пути. Таким образом регулируется глубина поиска.

1. Получив запрос повторно (от другого соседа), узел рассылает его лишь в том случае, если пакет способен пройти дальше остальных: его TTL меньше, чем у всех ранее проходивших через компьютер пакетов этого запроса [Петров, Тормасов, 2006; 2007].

Два этапа миграции. Будем предполагать, что распределенная система в целях повышения надежности регулярно производит следующие действия для каждого из виртуальных серверов:

- делает *слепок* (snapshot) сервера – файл, хранящий состояние виртуальной машины на текущий момент времени t_i^c ;
- записывает слепок в сеть;
- удаляет слепок для предыдущего момента времени t_{i-1}^c .

Для миграции такого виртуального сервера в некоторый момент t будет достаточно:

- собрать на новой машине его слепок по последний момент времени t_i^c ;
- скопировать с прежней машины на новую файл с разницей состояний виртуального сервера в моменты t_i^c и t .

Оценка длительности миграции

Задачу оценивания длительности поиска решим двумя способами в рамках различных моделей. Это позволит сравнить результаты и способы решения.

Имитационная модель распределенной системы. Распределенную систему будем понимать как граф, узлы которого есть компьютеры, хранящие некоторые данные, а ребра – сетевые каналы передачи информации. Каждый элемент графа, узел или ребро, характеризуется определенным набором параметров, например, производительностью машины или пропускной способностью узла. Отметим, что данные через постоянное сетевое соединение могут передаваться по нескольким физическим участкам сети, соединенным коммутаторами и маршрутизаторами.

Многие факторы в модели носят случайный характер, поэтому при описании происходящего была выбрана вероятностная модель. В ней параметры узлов и соединений системы представляются случайными величинами:

- Θ_n^1 – время пересылки сетевого пакета узлом n ;
- Θ_n^2 – время обработки запроса (на сборку файла) на узле n ;
- V_{nm} – пропускная способность соединения между узлами n и m ;
- Q_n – количество требуемых порций на машине n ;
- $\Psi_{nm} = \begin{cases} 1, & \text{если посылка пакета от } n \text{ до } m \text{ была} \\ & \text{успешной,} \\ 0, & \text{если при посылке пакета от } n \text{ до } m \\ & \text{произошла ошибка (пакет не дошел до } m \text{).} \end{cases}$

Заметим, $P\{\Psi_{nm} = 1\} = 0$ означает, что узлы n и m просто не соединены. Длительность миграции также представляется в модели случайной величиной, зависящей от вышеописанных.

Остальные параметры при оценке длительности миграции считаются заданными постоянными величинами:

- структура распределенной системы: множество узлов и схема связей между ними;
- τ_{ij} – сетевая задержка (latency) при передаче данных между узлами i и j , фактически определяется физическим расстоянием между узлами, поэтому меняться не может;
- d – глубина поиска (максимальное количество узлов, которое может пройти запрос с поиском).

В работе требуется найти функцию распределения длительности процесса миграции.

Передача данных между соседями. Стандартное представление времени передачи данных объема V по соединению между узлами i и j (обозначим его за $\Theta_{ij}^3(V)$) с пропускной способностью V_{ij} и задержкой τ_{ij} таково:

$$\Theta_{ij}^3(V) = \frac{V}{V_{ij}} + \tau_{ij},$$

т. е. длительность передачи данных между соседними узлами – это случайная функция, зависящая от случайной величины V_{ij} и неслучайного параметра V .

После введения $\Theta_{ij}^3(V)$ следует уточнить роль V_{ij} – она моделирует среднюю пропускную способность соединения между i и j за период передачи данных.

Пусть теперь узлы i и j не являются соседними. Рассмотрим множество путей из i в j $P_{ij} = \{\vec{l}_r : \vec{l}_r = (l_0 = i, l_1, \dots, l_{r-1}, l_r = j)\}$, состоящее из цепочек узлов, являющихся соседями \vec{l}_r . Определим $\Theta_{ij}^2(V)$ следующим образом:

$$\Theta_{ij}^2(V) = \min_{\vec{l}_r \in P_{ij}} \left\{ \sum_{k=0}^{r-1} (\Theta_{l_k l_{k+1}}^3(V) + \Theta_{l_{k+1}}^1) \right\}. \quad (1)$$

Утверждение 1: минимум в (1) достигается на элементе множества P_{ij} .

Доказательство: разделим все множество путей P_{ij} на два подмножества – пути, содержащие циклы (P_{ij}^1) и не содержащие (P_{ij}^2). Любой путь из P_{ij}^1 длиннее соответствующего ему пути из P_{ij}^2 , поэтому:

$$\min_{\vec{l}_r \in P_{ij}} \left\{ \sum_{k=0}^{r-1} (\Theta_{l_k l_{k+1}}^3(V) + \Theta_{l_{k+1}}^1) \right\} = \min_{\vec{l}_r \in P_{ij}^2} \left\{ \sum_{k=0}^{r-1} (\Theta_{l_k l_{k+1}}^3(V) + \Theta_{l_{k+1}}^1) \right\}.$$

Любой путь можно записать последовательностью узлов, через которые он проходит. Он содержит цикл, если в последовательности хотя бы один узел повторяется. Граф содержит конечное число узлов, поэтому количество последовательностей из неповторяющихся номеров узлов также конечно (не больше $\sum_{i=0}^S C_S^i \cdot i!$ для S узлов, поскольку $C_S^i \cdot i!$ – верхняя оценка для количества путей из i узлов). Таким образом, минимум по множеству P_{ij}^2 (а значит, и P_{ij}) достигается на одном из его элементов.

Пусть минимум из (1) достигается на некоторой цепочке $\vec{l}_r^* = (l_0^*, \dots, l_r^*)$ (эта цепочка – случайный вектор). Тогда успех / неудачу передачи данных между i и j будет описывать величина

$$\Psi_{ij}(\vec{l}_r^*) = \prod_{k=0}^{r-1} \Psi_{l_k^* l_{k+1}^*}.$$

Длительность миграции. Обозначим искомую длительность миграции за T_M . А длительности копирования на новую машину базовой части виртуального сервера из сети и файла с разностью состояний за T_B и T_D соответственно (индексы от слов *Base* – базовый и *Difference* – разница).

Тогда

$$T_M = \max \{T_B, T_D\},$$

поскольку этапы миграции выполняются одновременно.

Пусть V_D – объем файла с разностью состояний. Его копирование на новую машину займет время:

$$T_D = \Theta_{ij}^3(V_D),$$

где i и j – узлы источника и приемника процесса миграции соответственно.

Пусть V_B – объем базовой части виртуального сервера, а n_1, n_2, \dots, n_N – узлы, хранящие порции базовой части виртуального сервера. Поскольку размер одной порции в $N - k$ схеме равен $\frac{V_B}{k}$, то порции с узла n_i придут на целевой узел миграции c через время:

$$T_i = \Theta_{cni}^3(V_q) + \Theta_{ni}^2 + \Theta_{nic}^3\left(\frac{V_B}{k}\right), i = 1 \dots N. \quad (2)$$

Тогда длительность второго этапа миграции T_B равна k -му по порядку элементу последовательности (T_1, T_2, \dots, T_N) .

Вычислять функцию распределения напрямую по формуле (2) чрезвычайно сложно главным образом потому, что величины $\Theta_{cni}^3(V_q)$ оказываются зависимыми. Поэтому для решения задачи был выбран метод имитационного моделирования. В нем процесс миграции симулируется в соответствии в описанной моделью:

- сначала порции базовой части виртуального сервера «разбрасываются» по узлам распределенной системы (фактически генерируются значения Q_n);
- пересылка данных между соседями выражается в генерации значений V_{nm} и Ψ_{nm} ;
- значения Θ_n^1 и Θ_n^2 генерируются при пересылке данных или обработке запроса на поиск частей файла соответственно.

Заметим, что возможность неудачной пересылки данных (т. е. их потери) позволяет всему процессу миграции завершиться неуспешно. Результатом имитационного моделирования будет значение длительности миграции в случае успешного завершения миграции или просто сам факт неудачной миграции. Повторив процесс многократно, получим множество значений длительности (рис. 1) – выборку и процент успешного завершения операции. А выборка дает возможность построить эмпирическую функцию распределения случайной величины.

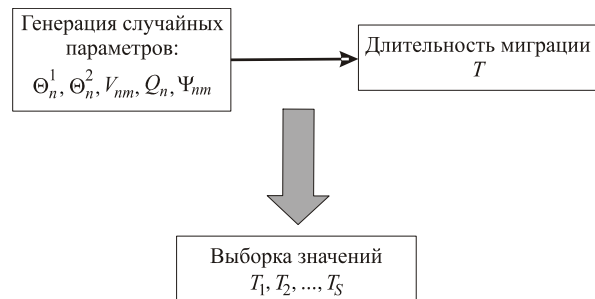


Рис. 1. Схема моделирования

Утверждение 2: описанный метод имитационного моделирования позволяет получить решение, сколь угодно близкое по вероятностной мере к истинному.

Доказательство: точная имитация процесса миграции в терминах модели распределенной системы гарантирует, что полученный в результате ряд значений действительно реализации длительности. А теорема Гливленко доказывает сходимость по вероятности ее эмпирической функции распределения к истинной.

Теоретическая модель распределенной системы. Для того чтобы оценить правдоподобность получаемых имитационным моделированием результатов, вычислим распределение длительности миграции T_M другим способом: в рамках новой модели. Вторая модель также использует теорию вероятности, однако:

- длительности пересылки и обработки пакетов, пропускные способности соединений постоянны (фактически заменяются на средние значения соответствующих случайных величин предыдущей модели):

$$\begin{aligned}\Theta_n^1 &\rightarrow \theta_n^1 = E(\Theta_n^1) = const; \\ \Theta_n^2 &\rightarrow \theta_n^2 = E(\Theta_n^2) = const; \\ V_{nm} &\rightarrow v_{nm} = E(V_{nm}) = const;\end{aligned}$$

- пересылка пакета всегда успешна:
 - $\Psi_{nm} = 1 \quad \forall(n, m)$;
- порции файла равномерно разбрасываются по узлам распределенной системы.

Представления других величин оставим неизменными: случайная величина Q_n и константы τ_{ij} и d , а также длительности двух этапов миграции T_B , T_D и объемы V_B , V_q и V_D . Упрощение модели необходимо для определения распределения длительности ввиду сложностей, описанных выше.

2. Пусть в распределенной системе M узлов. Вероятностный характер новой модели придает разброс порций по узлам сети. Он будет производиться генерацией значений слу-

чайного вектора $\Phi^N = \begin{pmatrix} \Phi_1 \\ \dots \\ \Phi_N \end{pmatrix}$, где компоненты $\{\Phi_i\}_{i=1}^N$ – одинаково распределенные случайные

величины, принимающие значения $1, \dots, M$ с равной вероятностью $(\frac{1}{M})$. Значение величины

Φ_i означает номер узла, на котором лежит i -я порция файла. Определим конкретнее вероятностное пространство вектора Φ^N [Чернова; Гнеденко, 1988]:

- множество элементарных исходов Ω – всевозможные векторы с целыми компонентами из $[1, M]$;
- алгебра событий – все подмножества Ω ;
- вероятностная мера – функция, определенная классическим образом.

Расчет длительности миграции. Длительности пересылки сетевого пакета объема V между узлами n и m θ_{nm} также будут постоянными величинами в новой модели:

$$\theta_{nm}^3(V) = \min_{l_r \in P_{ij}} \left\{ \sum_{k=0}^{r-1} \left(\frac{V}{v_{ij}} + \tau_{ij} + \theta_{l_{k+1}}^1 \right) \right\}.$$

Поскольку $T_M = \max\{T_B, T_D\}$, а $T_D = \theta_{ij}^3(V_D)$ – постоянная величина, то функция распределения T_M :

$$F_{T_M}(x) = P\{\max\{T_B, \theta_{ij}^3(V_D)\}\} = \begin{cases} 0, & x \leq \theta_{ij}^3(V_D), \\ F_{T_B}(x), & x > \theta_{ij}^3(V_D). \end{cases} \quad (3)$$

Найдем функцию распределения $T_B - F_{T_B}(x)$. Благодаря постоянству Θ_n^1 , Θ_n^2 и V_{nm} длительность сборки порций базовой части T_B здесь может принимать только дискретные значения, равные

$$T_n = \theta_{cn}^3(V_q) + \theta_n^2 + \theta_{nc}^3\left(\frac{V_B}{k}\right),$$

где $n = 1, \dots, M$, а c – целевой узел миграции.

Перенумеруем M узлов распределенной системы по возрастанию T_n :

$$T_1 \leq T_2 \leq \dots \leq T_i \leq \dots \leq T_M,$$

и сгруппируем рядом стоящие узлы с одинаковыми T_n в отдельные множества Q_1, \dots, Q_r , $r \leq M$:

$$Q_i = \{j \mid T_j = T_i'\},$$

$$\forall i = 1 \dots M-1, \forall j_1 \in Q_i, \forall j_2 \in Q_{i+1}: T_{j_1} < T_{j_2}.$$

Тогда вероятность для длительности сборки порций базовой части T_B принять значение T'_i , $P\{T_B = T'_i\}^r_{i=1}$, будет равна вероятности найти последнюю из необходимых, k -ю порцию файла на одном из узлов из множеств $Q_i^r_{i=1}$ соответственно.

Итак, в новой модели T_B – случайная величина, принимающая дискретный ряд значений T'_i с вероятностями $P\{T_B = T'_i\}$. Их совокупность и представляет собой распределение T_B . Найдем $P\{T_B = T'_i\}$.

Обозначим за S_i область поиска, покрываемую за T'_i : $S_i = \bigcup_1^i Q_j, i = 1 \dots r$. Пусть также, для общности, $Q_0 = S_0 = \emptyset$ (пустое множество). Рассмотрим набор событий

$$A_i = \bigcup_{j=0}^{k-1} (\{B S_{i-1} \text{ содержит } j \text{ порций}\} \cap \{Q_i \text{ содержит } \geq k - j \text{ порций}\}).$$

Докажем ряд утверждений относительно этих событий.

Утверждение 3: Все события A_1, A_2, \dots, A_r – независимы.

Доказательство: покажем, что $\forall i \forall q : q > i \ A_i \cap A_q = \emptyset$. Действительно, во всех элементарных исходах из множества $A_q, q > i$, узлы из $S_{q-1} = \bigcup_1^{q-1} Q_j$ содержат не более $k - 1$ порций.

Поскольку $q > i$, то $S_i \subseteq S_{q-1}$. Но множество A_i содержит только такие элементарные исходы, для которых $S_i = S_{i-1} \cup Q_i$ содержит не менее k порций. Поэтому A_i и A_q никак не могут иметь общих элементарных исходов, т. е. не пересекаются.

Утверждение 4: $A_1 \cup A_2 \cup \dots \cup A_r = \Omega$ (Ω – пространство всех элементарных исходов).

Доказательство: для любого элементарного исхода из Ω (разброса N порций среди всех узлов) рассмотрим количество порций на узлах из множеств S_i . Поскольку $S_{i-1} \subset S_i \ \forall i = 2, \dots, r$, то S_i содержит не меньше порций, чем S_{i-1} . Самое крупное множество узлов S_r содержит все N порций. Поэтому всегда найдется такой номер $2 \leq i \leq r$, для которого S_{i-1} содержит все еще не более $k - 1$ порций, а S_i содержит уже не менее k порций. Это как раз и означает, что выбранный нами элементарный исход принадлежит A_i . Иными словами,

$$\forall \omega \in \Omega \ \exists i : \omega \in A_i,$$

что и требовалось доказать.

Утверждение 5: длительность сборки порций базовой части T_B может быть равна T'_i в том и только том случае, если в S_{i-1} содержатся $j \leq k - 1$ порций искомого файла и Q_i содержит не менее $k - j$ порций, т. е. идентичны события $\{T_B = T'_i\} = A_i$.

Доказательство: поскольку для сборки файла необходимо не менее k порций, то условие «в S_{i-1} содержится $j \leq k - 1$ порций искомого файла» гарантирует, что поиск не закончится раньше T'_i , при этом « Q_i содержит не менее $k - j$ порций» обязывает его закончиться на одном из узлов множества Q_i . Длительность в этом случае будет равна T'_i . И наоборот, $T_B = T'_i$ означает, что поиск завершился на одном из узлов Q_i , не раньше.

Утверждение 6: вероятности $P(A_i)$ определяются формулами

$$P(A_1) = \sum_{j=k}^N C_N^j \left(\frac{|Q_1|}{M}\right)^j \left(1 - \frac{|Q_1|}{M}\right)^{N-j}, \quad (4)$$

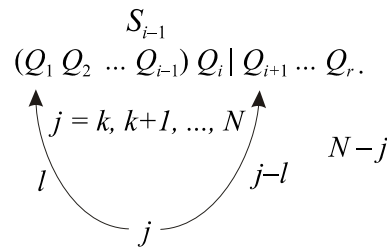
$$P(A_i) = \sum_{j=k}^N C_N^j \left[\sum_{l=0}^{k-1} C_j^l \left(\frac{|S_{i-1}|}{M}\right)^l \left(\frac{|Q_i|}{M}\right)^{j-l} \right] \left(1 - \frac{|S_{i-1}| + |Q_i|}{M}\right)^{N-j}, \quad 2 \leq i \leq r-1, \quad (5)$$

$$P(A_r) = \sum_{l=0}^{k-1} C_N^l \left(\frac{|S_{r-1}|}{M}\right)^l \left(\frac{|Q_r|}{M}\right)^{N-j}. \quad (6)$$

Здесь C_N^k – биномиальный коэффициент.

Доказательство: покажем, откуда берется формула (5). Формулы (4) и (6) лишь ее частные случаи, расписывающие неопределенности типа 0^0 : ведь $|S_0| = (1 - \frac{|S_{r-1}| + |Q_r|}{M}) = 0$. Они получатся из (5), если считать, что $0^0 = 1$.

В формуле (5) для $P(A_i)$ по j суммируются вероятности разбить все множество узлов на две группы: $S_i = S_{i-1} \cup Q_i$ и остальные узлы так, чтобы S_i содержало $j \geq k$ порций. Причем эти j порций должны быть распределены следующим образом: $l \leq k-1$ порций в множестве S_{i-1} и остальные $j-l$ порций в множестве Q_i , за что и отвечает внутреннее суммирование по l . Итак распределение порций в случае события A_i выглядит следующим образом:



Таким образом, утверждения 3–5 позволяют нам искать $P(A_i)$ вместо $P\{T_B = T_i^1\}$, а утверждение 6 дает численные выражения для них. Кратчайшие времена достижения узлов $\theta_{cn}^3(V)$ вычислим по алгоритму Дейкстры. Распределение длительности миграции T_M полностью определяется формулами (3)–(6).

Пример

Выбор распределений. Расположение порций базовой части по узлам распределенной системы при расчете по обеим моделям считается в примере равномерным. На одном компьютере при этом могут располагаться хоть все N порций (с вероятностью $(\frac{1}{M})^N$). Мы исходим из предпосылки, что мы не имеем никакой информации о том, где лежат эти порции. Например, они могут копироваться с машины на машины по специальному алгоритму для повышения характеристик системы, т. е. фактически при расчетах по обеим моделям генерируются значения случайного вектора Φ^N из раздела *модель распределенной системы*. В имитационной модели они определяют значения величины Q_n .

Распределения времени обработки (Θ_n^2) и пересылки (Θ_n^1) запроса узлом n принимаются нормальными ($\in N(1000 \text{ мс}, 100 \text{ мс})$). Такой вид распределения – самый распространенный в природе, так что выбор представляется естественным. Значения математического ожидания и дисперсии было взято из опытных данных по обработке сетевых запросов.

Пропускная способность V_{nm} соединения между n и m в имитационной модели распределена равномерно на отрезке $[1, V_{nm}^{\max}]$, чтобы подчеркнуть спонтанный, случайный характер сети. Параметр V_{nm}^{\max} был выбран равным очень распространенному в современных локальных сетях значению пропускной способности в $100 \frac{\text{Мбит}}{\text{с}}$ или $13107,2 \frac{\text{байт}}{\text{мс}}$. В теоретической модели пропускная способность равна $\frac{V_{nm}^{\max}}{2}$.

Вероятность успешной передачи данных между соседними узлами была принята равной 0,95, т. е. величина $\Psi_{nm} \in B_{0,95}$.

Распределенная система. Структура распределенной системы, на которой производились все расчеты, представлена на рис. 2. Для хранения файлов используется $N - k$ -схема с $N = 10, k = 5$. Исследуется миграция с узла № 19 на узел № 5. При этом на узле 5 собирается базовая часть мигрируемого виртуального сервера. Одновременно файл с разницей состояний переносится по выделенному пути. Глубина поиска порций базовой части равна 3.

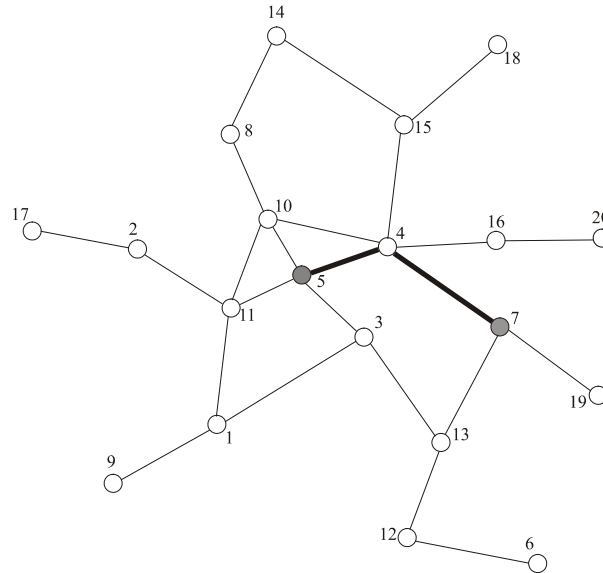


Рис. 2. Структура распределенной системы

Объяснение отличий. На рис. 3 представлены функции плотности (слева) и распределения (справа) длительности миграции в описанных выше условиях моделирования. Для теоретической модели множество всех точек плотности вероятности различимо разделено на три группы. Первая (до $2,2 \cdot 10^4$ мс) соответствует расположению k порций базовой части на узле 5 и его ближайших соседях (узлах 3, 4, 10, 11). Вторая ($3,5 \cdot 10^4 - 4,0 \cdot 10^4$ мс), самая вероятная, – расположению k порций на соседях «второго порядка» узла 5 (узлах 1, 2, 7, 8, 13, 15, 16). И третья (более $5,3 \cdot 10^4$ мс) – на самых дальних узлах.

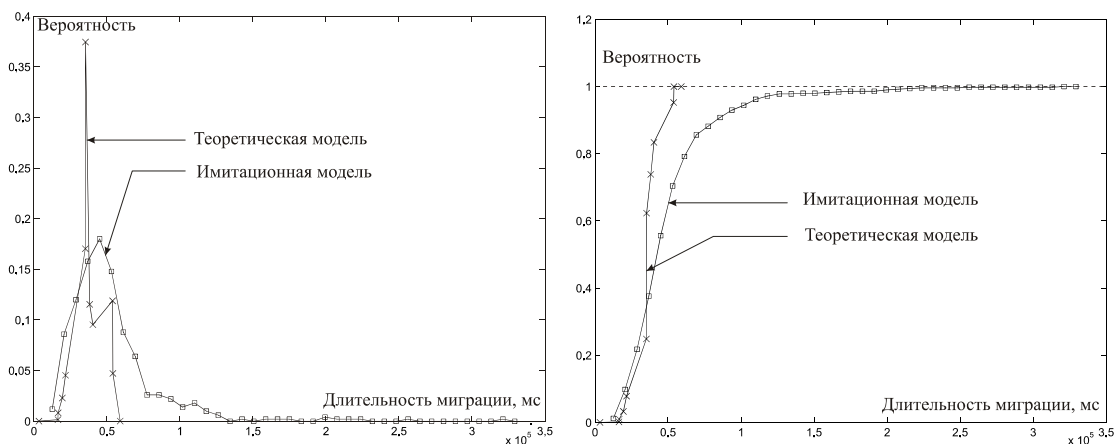


Рис. 3. Распределение длительности миграции в разных моделях

В имитационной модели за счет учета дополнительных случайных параметров нет различий между группами. Более того, наблюдается интересный феномен: она дает большую в среднем длительность поиска (математические ожидания равны $3,8 \cdot 10^4$ мс для теоретической модели и $5,2 \cdot 10^4$ мс для имитационной). Он возникает из-за того, что длительность передачи данных $\Theta_{ij}^3(V)$ обратно пропорциональна случайной пропускной способности V_{ij} .

Например, рассмотрим две последовательные пересылки данных от узла A через B к C . Пусть для простоты $\Theta_{ij}^3(V) = \frac{V}{V_{ij}}$.

1. Если пропускная способность при обеих пересылках постоянна $V_{ij} = V^*$, то

$$\Theta_{AC}^3(V) = \frac{V}{V_{AB}} + \frac{V}{V_{BC}} = 2 \frac{V}{V^*}.$$

2. Если пропускная способность равномерно распределена на некотором отрезке (как в нашей имитационной модели) и равна сначала $V_{AB} = \frac{V^*}{2}$, а потом $V_{BC} = \frac{3V^*}{2}$, то

$$\Theta_{AC}^3(V) = \frac{V}{V_{AB}} + \frac{V}{V_{BC}} = 2 \frac{2}{3} \frac{V}{V^*}.$$

Таким образом, в использованных условиях моделирования соответствия в распределениях быть не может.

Сводимость имитационной модели к теоретической. Покажем, что при изменении отрезка распределения пропускной способности V_{nm} :

$$[1, V_{nm}^{\max}] \rightarrow \left[\frac{1}{r}, 1 - \frac{1}{r}\right] \cdot V_{nm}^{\max},$$

средняя длительность поиска имитационной модели приближается к теоретической при $r \rightarrow 2$ (рис. 4).

Графики функций плотности и распределения длительности миграции в конечной точке, при $r = 2$, показаны на рис. 5. Нетрудно заметить, что четкое разделение трех описанных выше групп точек появилось и в имитационной модели.

Заметим, что полное совпадение распределений невозможно из-за коренных различий моделей: длительность миграции в теоретической модели дискретна. Ее распределение есть набор значений с вероятностями их принимать. Количество значений равно числу узлов системы (20). В имитационной модели строится эмпирическая функция распределения, которая может быть непрерывной. Причем подробность деталей распределения определяется количеством точек (100 для графиков на рис. 5).

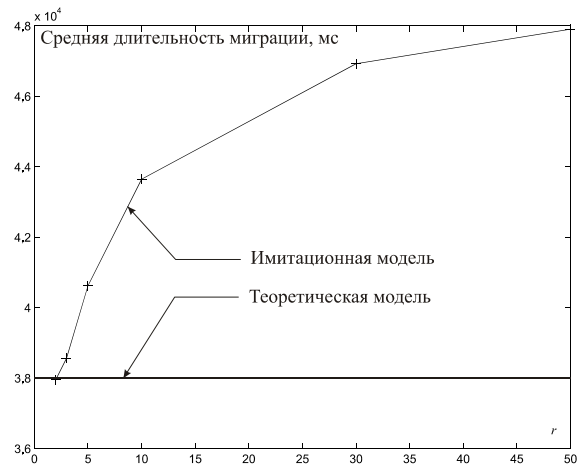


Рис. 4. Средняя длительность миграции

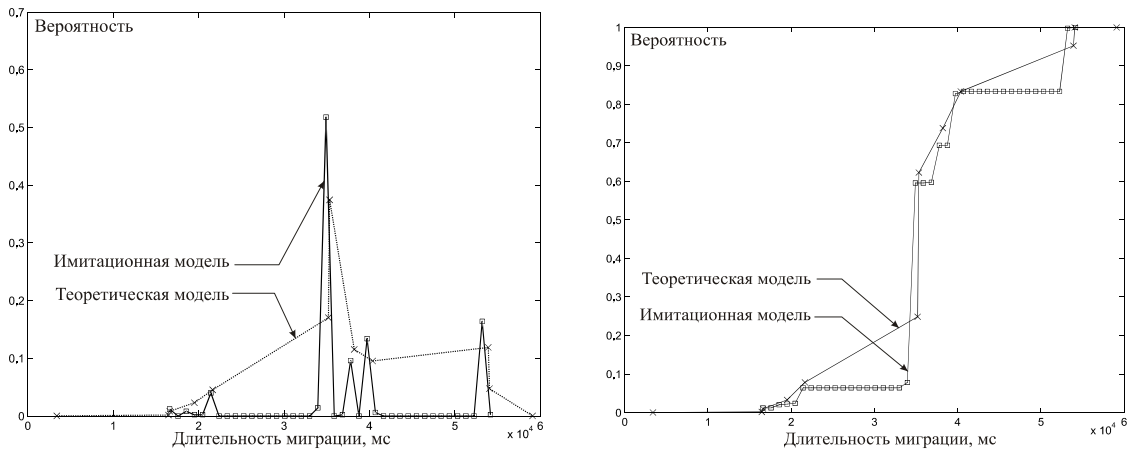


Рис. 5. Распределение длительности миграции при постоянной пропускной способности в имитационной модели

Проще говоря, график теоретического распределения должен быть более грубым, усредненным, что мы и наблюдаем на рис. 5.

Преимущества двухэтапной миграции. Длительность миграции определяется двумя величинами: длительностью сборки базовой части и временем переноса файла с разностью состояний. В приведенных выше результатах размер файла с разностью состояний был равен 10 Мб, против 500 Мб базовой части. Поэтому вторая величина вносила в длительность миграции слабый вклад. На рис. 6 показано изменение этого вклада при изменении размера файла с разностью состояний. По оси Y отложен процент точек, в которых $\max\{T_B, T_D\} = T_D$.

При размерах файла с разницей состояний, больших 180 Мб, его копирование на целевую машину вносит основной вклад в длительность всего процесса. Это подчеркивает эффективность двухэтапной схемы миграции по сравнению с обычным переносом всего виртуального сервера с текущего компьютера на целевой. Моделирование позволяет также выбрать оптимальный размер файла с разницей состояний (он определяется частотой обновления базовой части виртуального сервера в системе).

Практическая ценность результатов. Предложенные теоретическая и имитационная модели процесса миграции имеют важное практическое значение.

1. Проектирование распределенных систем не может обойтись без их моделей. Получить аналогичные результаты исходя из эксперимента – гораздо более затратный способ.

2. Функция распределения позволяет спрогнозировать длительности миграции с любой заданной вероятностью. И, наоборот, для любой целевой длительности миграции можно оценить вероятность, с которой она будет превышена.

3. По функции распределения можно вычислить среднее значение длительности, а значит, и его зависимость от изменения входных параметров. Например, от объема базовой части мигрируемого сервера, который определяется частотой ее записи. Построенный график покажет, как достичь целевых значений длительности миграции.

Заключение

Задача о длительности миграции виртуального сервера в распределенной системе решена в работе в рамках двух математических моделей. В первой полностью имитируется процесс миграции, строится выборка ее длительности, как случайной величины. По выборке строится эмпирическая функция распределения. Во второй для длительности миграции теоретически рассчитываются вероятности принимать ряд конкретных значений. Эти вероятности фактически и образуют плотность распределения длительности.

Реализация обеих моделей продемонстрирована на конкретном примере. Результаты показали различие применяемых моделей и сводимость одной к другой при определенных условиях. В целом более предпочтительно выглядит имитационная модель. Она гибче, учитывает больше случайных параметров, и поэтому ближе к реальной ситуации.

Помимо прочего, моделирование позволяет проследить зависимость результатов миграции от входных параметров и на основе этой информации выбирать их оптимальные (для минимизации длительности миграции) значения.

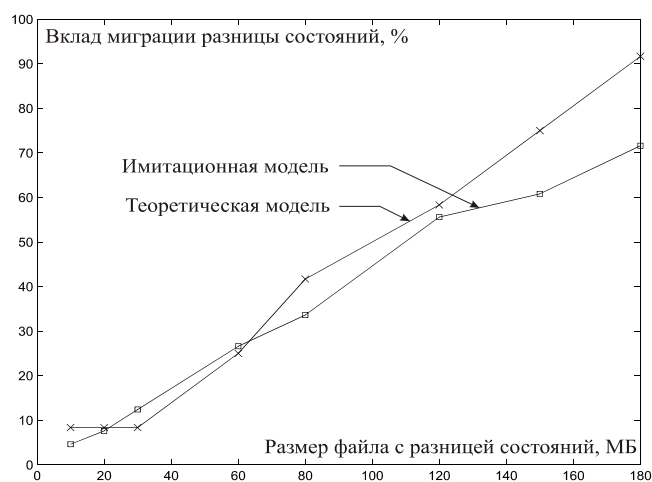


Рис. 6. Вклад миграции файла с разницей состояний в общую длительность

Список литературы

Хасин М. А. Модель распределенного хранилища в глобальной сети: Работа на соискание степени канд. физ.-мат. наук. М.: МФТИ, 2001.

Castro M., Druschel P., Kermarrec A., Nandi A., Rowstron A., Singh A. SplitStream: High-Bandwidth Multicast in Cooperative Environments. Microsoft Research, Rice University, 2004 [Электронный ресурс]. Режим доступа: <http://research.microsoft.com/~antr/PAST/SplitStream-sosp.pdf>.

Петров В. А., Тормасов А. Г. Доставка информации пользователю в распределенных системах // Проблемы вычислительной математики, математического моделирования и информатики: Сб. науч. тр. М.: МФТИ, 2006.

Петров В. А., Тормасов А. Г. Влияние пропускной способности соединений на длительность поиска в регулярных графах // Моделирование процессов обработки информации: Сб. науч. тр. М.: МФТИ, 2007.

Чернова Н. И. Теория вероятностей: Курс лекций. [Электронный ресурс]. Режим доступа: <http://www.nsu.ru/mmftvims/chernova/tv/portr.pdf>.

Гнеденко Б. В. Курс теории вероятностей. М.: Наука, 1988.

Pinheiro E. Truly-Transparent Checkpointing of Parallel Applications // Federal University of Rio de Janeiro, UFRJ., 2002 [Электронный ресурс]. Режим доступа: <ftp://ftp.inria.fr/INRIA/publication/publi-ps-gz/RR/RR-5755.ps.gz>.

Материал поступил в редколлегию 21.08.2008

V. A. Petrov, A. G. Tormasov, A. L. Mirkin

VIRTUAL SERVERS MIGRATION DURATION IN DISTRIBUTED SYSTEMS

A problem of virtual server migration duration in distributed systems is resolved in the article. Solution is made on the base of two models. Models are implemented programmatically. Results are demonstrated at the concrete example.

Keywords: virtual server, $N-k$ schema, file assemble, search query, search depth, simulation, probabilistic model, sample, search area, random vector, event algebra, probability measure, reference set, average of distribution, binomial coefficient, Dijkstra's algorithm.