ISSN 2079-3316 **ПРОГРАММНЫЕ СИСТЕМЫ: ТЕОРИЯ И ПРИЛОЖЕНИЯ** **т. 13**, № 4(55), с. 3–26
научная статья ПРОГРАММНОЕ И АППАРАТНОЕ ОБЕСПЕЧЕНИЕ ДЛЯ СУПЕРЭВМ

УДК 004.272.45
10.25209/2079-3316-2022-13-4-3-26;

Синхронное выполнение групповых операций в распределенных компонентах суперкомпьютеров и компьютерных кластерах

Геннадий Георгиевич **Стецюра**
Институт проблем управления имени В. А. Трапезникова РАН
(*подробнее об авторе на с. 23*)

Аннотация. В статье предлагаются децентрализованные процессы синхронизации действий распределенной группы активных компонентов (объектов) в суперкомпьютерах и компьютерных кластерах, ускоряющие их синхронный переход в заданные состояния и воздействие на внешнюю среду. Объектам не задается значение момента времени перехода. Им известен только факт появления совокупности событий, требующих наиболее быстрого перехода всех объектов в новое состояние. Для синхронизации объекты обмениваются оптическими сигналами по беспроводным каналам связи через ретранслятор оптических сигналов, объединяющий группы объектов. Синхронизация ускоряется за счет применения групповых операций, каждая из которых *одновременно* получает и обрабатывает данные группы распределенных объектов. Такая обработка выполняется групповыми операциями при передаче данных объектами, не увеличивая это время. Причем время обработки не зависит от количества данных, одновременно обрабатываемых операцией. Групповые операции выполняются в ретрансляторе, не содержащем вычислительных средств. В целом решения статьи ускоряют при возникновении непредвиденных событий переход асинхронно действующих распределенных объектов в заданное синхронное состояние. Такая возможность наиболее востребована для систем, работающих в режиме жесткого реального времени. (*see abstract in English on p. 24*)

Ключевые слова и фразы: суперкомпьютеры, компьютерные кластеры, децентрализованное управление, многоуровневая синхронизация действий объектов, распределенные внутрисетевые вычисления, групповые операции

Для цитирования: Стецюра Г. Г. *Синхронное выполнение групповых операций в распределенных компонентах суперкомпьютеров и компьютерных кластерах* // Программные системы: теория и приложения. 2022. **Т. 13**. № 4(55). С. 3–26. http://psta.psiras.ru/read/psta2022_4_3-26.pdf

Введение

В статье решается следующая задача. Имеется группа распределенных компонентов (объектов) суперкомпьютера (СК) или компьютерных кластеров. Взаимодействие между объектами осуществляется по беспроводным оптическим каналам связи. Объекты группы неожиданно обнаруживают возникновение события, требующего наиболее быстрого их совместного перехода в новое состояние. Объекты должны как можно быстрее информировать всю группу о возникновении этого события и организовать одновременный переход группы в требуемое состояние.

Операции перехода должны выполняться децентрализовано. Это одно из требований для получения высокой скорости перехода, так как участие центра управления сопровождается дополнительными затратами времени.

Требование быстрой реакции на непредсказуемое событие исключает часто применяемое, но медленно реализуемое указание времени перехода в новое состояние с помощью часов. Вместо этого в поставленной задаче объекты ускорят синхронизацию, начиная ее после получения сигнала о возникновении события, требующего их согласованных действий.

Сетевые средства, необходимые для распределенной системы, существенно замедляют синхронизацию. Для уменьшения замедления в статье сетевые средства кроме передачи сообщений выполняют ряд дополнительных операций по обработки данных без привлечения компьютеров.

Получить указанный в задаче результат известными способами не удалось, и автором были разработаны новые способы и выполняющие их операции. Часть их была опубликована ранее и адаптирована к предлагаемой задаче, часть предлагается в статье впервые. Основные из них — групповые операции и децентрализованный способ синхронизации действий объектов. Для их выполнения требуется приведенная ниже организация структуры связей между объектами.

Под групповой операцией понимается элементарная операция, которая одновременно получает и обрабатывает данные, поступающие от группы объектов СК. Групповые операции выполняются объектами при получении ими одной из таких операций — групповой команды, которую рассылает всем объектам один или несколько из них.

Выполняется групповая операция в процессе передачи данных объектами, не увеличивая это время. Время обработки не зависит от количества данных, одновременно обрабатываемых операцией.

Групповые операции в статье выполняются преимущественно в средствах сети, объединяющей объекты, в основном, в ее простом, не содержащем логических элементов, ретрансляторе сигналов. Ретранслятор также выполняет важную, центральную для статьи функцию. Он заменяет группы произвольно расположенных в пространстве источников и приемников сигналов одним источником и приемником. Только при такой замене стало возможным выполнять групповые операции с участием произвольно расположенных относительно друг друга источников и приемников сообщений.

Таким образом, здесь сетевые средства, не содержащие компьютеры, одновременно осуществляют передачу сообщений, контролируют процесс передачи и выполняют распределенные вычисления. Отсутствие программной обработки данных также позволило повысить скорость выполнения групповых операций. Предлагаемые действия по связи между компьютерами, следуя общепринятой практике, не должны требовать каких-либо изменений в структуре компьютера. Так как предлагаемые средства достаточно просты, то смогут выполняться вне компьютера в специализированной сетевой карте.

В статье рассматриваются однослойные и многослойные групповые операции. При однослойной операции все объекты-приемники образуют одну группу, которая получает общую для нее групповую команду. В многослойной групповой операции объекты делятся на упорядоченные слои. В первом слое приемники слоя одновременно выполняют команды источника. Во втором и последующих слоях приемники предыдущего слоя становятся источниками данных и команд для приемников следующего слоя. В каждой такой паре слоев обеспечивается одновременное выполнение команд приемниками. В системе могут одновременно существовать несколько многослойных операций, состав объектов в слоях и операциях формируется в динамике.

Следует отметить, что и ранее сетевые средства выполняли не только передачу сообщений, но также управление использованием сети и выполнение в ней вычислений. В 1997 г. в статье [1] предложено использовать сетевые компьютерные средства для управления трафиком, затем появились работы, предлагающие использовать такие средства для управления конфигурацией сети [2], появилось направление

In-Network Computing [3–5], в котором ряд видов вычислительных операций также выполняют сетевые компьютеры. В отличие от этих работ для повышения реактивности систем предлагается, как отмечено выше, не использовать в сети компьютеры. Их заменили не содержащие вычислительных средств ретрансляторы.

Для предлагаемых ниже решений необходимо точно определять временной интервал прохождения сигналом расстояния между компонентами системы. Основой для таких измерений является промышленный сетевой протокол PTP, корректирующий показания часов объекта [6–8], и протокол, созданный в проекте White Rabbit для прецизионных физических экспериментов [9, 10]. В настоящее время разработан стандарт, объединяющий эти два метода [8]. Эти методы используют для выполнения измерений ведущий объект (лидер). Лидер имеет точные часы и поочередно синхронизирует показания часов ведомых объектов, измеряя расстояние объектов от лидера. В [9, 10] ошибка в измерениях времени лежит в субпикосекундном диапазоне. Эти результаты использованы без изменений в статье. Но лидер заменен одним или группой произвольных объектов.

Все решения статьи ориентированы на расстояния между объектами СК в десятки метров и менее. Ограничение расстояний необходимо для получения высокой реактивности распределенной системы.

Результаты статьи распределены по ее разделам следующим образом. В разделе 1 рассматривается структура объектов, выполняющих групповые операции. В разделе 2 приводятся методы синхронизации приемников групповых команд. В разделе 3 описана синхронизация источников команд. В разделе 4 рассмотрены особенности группового взаимодействия объектов, а также дано краткое описание групповых команд, не включенных в статью. Разделы описывают полный набор инструментов, необходимый для синхронизации действий объектов.

1. Средства для выполнения групповых операций

1.1. Структура связей между объектами

Средством выполнения многослойных групповых операций в СК является развитие структуры выполнения однослойных групповых операций в СК, предложенной автором ранее [11]. Ее фрагмент представлен на рисунке 1.

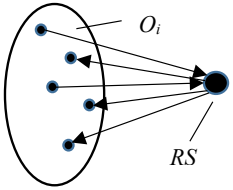
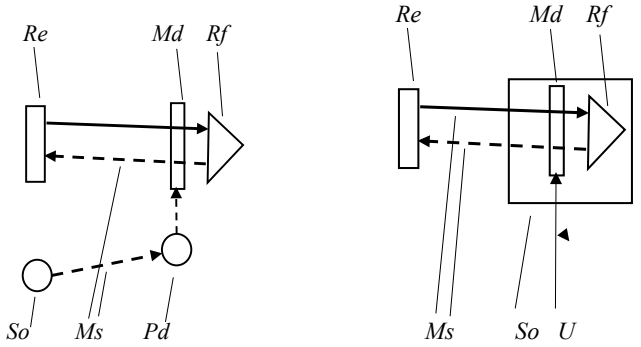


Рисунок 1. Сеть с единственным ретранслятором

На рисунке изображены объекты двух типов: объекты O_i — источники (группа So) и приемники (группа Re) сигналов и объект RS — ретранслятор оптических сигналов объектов O_i. Для приема сигналов от RS объекты Re посылают непрерывные оптические сигналы частот ${}^*f_1$ и ${}^*f_0$. Источник So посылает RS сигналы f_1 и f_0, несущие сообщения. RS модулирует сигналами сообщения непрерывные сигналы, возвращаемые на приемник. Объекты могут содержать переключатель для выбора необходимого RS.

Как показано на рисунке 2*а*, объект RS представляет собой оптический ретрорефлектор (R_f), который возвращает сигнал, поступающий на RS, в его источник. RS содержит фотоприемники (Pd) сигналов f_1



(*а*) новая схема (*б*) прежняя схема

Рисунок 2. Оптический ретранслятор сети

и f_0 и фильтры-модуляторы (Md) сигналов ${}^*f_1$ и ${}^*f_0$.

Фильтры обычно закрыты, и сигналы ${}^*f_1$ и ${}^*f_0$, поступающие в RS, не возвращаются к своим источникам. Когда сигнал f_1 или f_0 поступает на фотоприемники Pd, частотный фильтр ${}^*f_1$ или ${}^*f_0$ открывается, соответственно. Все приемники, посылающие сигналы ${}^*f_1$ или ${}^*f_0$, фиксируют приход сигналов f_1 и f_0 в RS от объектов-источников. Эти

действия определяют схему обмена сообщениями сигналов Ms между объектами O_i.

Эта структура обмена сообщениями развивает разработанный ранее в [**12**] метод получения данных от источника за счет энергии устройства, запрашивающего данные. Он показанна рисунке 2*б*. Приемник данных Re посылает непрерывный оптический сигнал на ретрорефлектор источника данных So. Последний возвращает приемнику свой сигнал, модулируя его содержанием запрашиваемых данных. Здесь нет отдельного объекта RS. Источник So имеет модулятор Md входящих сигналов опроса от Re. Фотодетекторы отсутствуют. Сигналы сообщения (Ms), поступающие в So, формируют ответные сигналы Re в Md. Этот метод был разработан для морских приложений. Предлагалось также использовать его для связи между спутниками [**13**]. Для простых систем связи этого достаточно, но СК требует дополнений, показанных на рисунке 2*а*, и коммутации связей между многими RS. Потребуется также особая синхронизация обмена сообщениями, обсуждаемая в других разделах этой статьи.

Система коммутации с многими ретрансляторами, использующая узлы рисунка 1 показанана рисунке 3. Здесь есть группы объектов O_i и

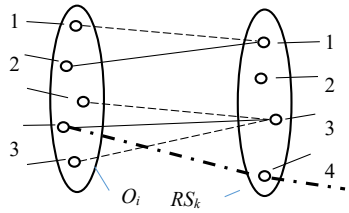


Рисунок 3. Структура коммутатора с многими ретрансляторами

RS_k. Значение k может начинаться от 1 и превышать n — количество объектов O_i. Каждый O_i должен иметь средства для отправки любому, определяемому текущими требованиями выполняемой задачи объекту RS_k сигналов f_1, f_0, ${}^*f_1$, ${}^*f_0$, и приема от любого RS_k сигналов ${}^*f_1$, ${}^*f_0$, модулированных сигналами f_1, f_0 других O_i.

Рассмотрим процесс коммутации в однослойной структуре рисунка 3.

Поскольку только RS_k является единственным посредником между источником и приемником сообщения, который не вносит задержку в процесс передачи, переключение является простым. Источник и получатели сообщения подключаются указанным выше

способом к заранее известному им RS_k, и происходит общая передача сообщения всей группе получателей. В то же время, используя другие RS_k, будет осуществляться передача различных источников. На рисунке 3 один из ретрансляторов группы RS_k (ретранслятор 4) будет необходим в качестве ${}^*RS_k$ для связи между группами объектов на рисунке 4. Для того чтобы передать сообщение одновременно всем O_i, целесообразно ввести широковещательный RS, который по ненаправленному оптическому или радиоканалу передает полученное им сообщение всем объектам.

Организация многоуровневой работы показанана рисунке 4.

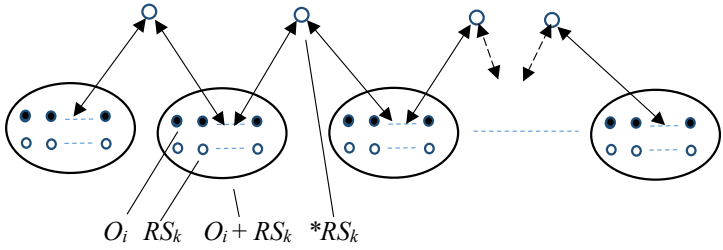


Рисунок 4. Многослойная структура связей

Здесь объекты O_i разделены на группы, отнесенные к разным слоям. Каждой такой группе присваивается группа RS_k. Внутри этой группы выделяется ретранслятор слоя ${}^*RS_k$.

Он отличается от остального слоя RS_k только тем, как он используется при взаимодействии слоев. К объекту ${}^*RS_k$ одновременно обращаются O_i двух слоев — текущего и предыдущего. Как было указано во введении, объекты O_i предшествующего слоя работают с объектами ${}^*RS_k$ как источники, а объекты O_i текущего слоя - как приемники. Переключение связей между объектами внутри слоя должно выполняться чаще и быстрее, чем переключение связей O_i с объектами ${}^*RS_k$. Поэтому эти виды переключения могут выполняться техническими средствами, существенно отличающимися по скорости.

Отметим основные возможности структур подраздела 1.1.

1. Группа объектов может использовать отдельный канал для каждого парного соединения (режим коммутатора) или общий канал для подключения источников к одному приемнику (режим шины). Оба типа структур могут существовать одновременно. Структура связей изменяется с высокой скоростью без снижения скорости обработки данных при посылке объектам групповой команды.

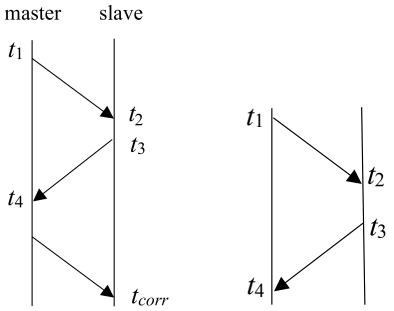
2. Между источниками и приемниками существует один промежуточный RS, который не снижает скорость взаимодействия между объектами и позволяет точно синхронизировать передаваемые данные при их поступлении на RS. Выравнивание прихода на RS одноименных битов сообщений группы источников служит основой для всех групповых управляющих и вычислительных операций, применяемых далее в статье.

3. Структура позволяет быстро заменить вышедший из строя канал связи. Достаточно иметь один резервный RS для замены любого отказавшего RS. Кроме того, как правило, каналы имеют запас по ширине полосы, и нагрузка отказавшего канала может быть перераспределена между ними.

1.2. Способы измерения расстояний между объектами

Здесь мы кратко изложим основные принципы измерения временных интервалов, необходимых оптическому сигналу для преодоления расстояния между двумя устройствами в методе PTP, приведенном во введении. Два объекта, ведущий и ведомый, взаимодействуют, как показано на рисунке 5*а*.

Ведущий посылает ведомому в момент t_1 сигнал о начале синхронизации и отсчет своих часов. Эта информация в момент t_2 поступает к ведомому. Ведомый корректирует свои часы по часам ведущего.



(*а*) RS_k (*б*) O_j

Рисунок 5. Измерение времени в PTP и вариант его использования в статье

В момент t_3 ведомый посылает ведущему ответный сигнал и значение своих часов. Ведущий в момент t_4 получает этот ответ, определяет время передачи сигнала между ведущим и ведомым $\tau = (t_4$

$-t_3)/2$ и сообщает ведомому значение τ скорректированного показания часов. Ведомый корректирует показания часов. Если не вдаваться в некоторые подробности, то это основа коррекции времени в PTP.

Используемый в статье вариант измерения без использования часов показан на рисунке 5*б*. Измеряется время передачи сигнала между объектом и ретранслятором сигналов RS. Для оптических сигналов в качестве RS может быть использован пассивный ретрорефлектор. Произвольный объект O_i имеет таймер, включает его и в момент времени t_1 посылает сигнал на RS. Сигнал поступает в RS в момент времени t_2. С задержкой в RS в момент времени t_3 сигнал будет отправлен от RS, и в момент времени t_4 объект O_i определяет время передачи сигнала между O_i и RS как $T_{OiRS} = ((t_4 - t_1) - - - (t_3 - t_2))/2$.

WR позволил добиться более точного способа измерения временных интервалов. Простыми средствами была достигнута точность в фемтосекундном диапазоне [**10**]. WR использует цифровой вариант способа биений.

В данной статье последний метод наиболее полезен для синхронизации объектов, выполняющих высокоскоростные совместные распределенные вычисления.

2. Синхронизация действий приемников команд

В этом разделе описывается синхронизация распределенных объектов RS для выполнения требуемых действий одновременно или с дополнительными временными задержками, которые могут быть заданы для каждого объекта. В подразделе 2.1 рассматривается одноуровневая синхронизация приемников команд, посылаемых одним источником. В подразделе 2.2 представлена многоуровневая синхронизация с одним источником команд.

2.1. Одноуровневая синхронизация приемников команд

Пусть имеется группа источников команд, из которых только один, произвольно расположенный O_i, посылает команду объектам-приемникам через RS_k. Пусть приемникам команд известно расстояние от RS_k и время передачи сигнала между этими объектами. Для того чтобы группа действовала одновременно, каждый член группы, получив команду, должен задержать свое действие, чтобы уравнять свое расстояние от RS_k с расстояниями других объектов.

Для этого объект O_i после получения команды выполняет задержку $D_i = T - T_i$. Здесь T - интервал времени, не менее времени передачи

сигнала между RS_k и самым дальним от RS_k приемником, T_i - интервал времени передачи сигнала между RS_k и O_i. Легко проверить, что при таком D_i действия команд повторных приемников будут выполняться одновременно. Если к D_i для O_i добавить задержку a_i, то a_i сдвинет действие O_i относительно других приемников. Это действие завершает одноуровневую синхронизацию.

Выполнение синхронных действий значительно усложняется, когда происходят изменения в динамике времени T_i. Существует две основные причины таких изменений — изменение расстояний между RS_k и O_i из-за внешних воздействий, например, изменения температуры, и из-за присутствия подвижных объектов, например, при включении СК в систему управления. Если длительность сигналов, передаваемых между RS_k и O_i, превышает T, то этими изменениями можно пренебречь. Тем не менее, в СК, при работе с сигналами пикосекундной длительности, изменения T_i необходимо учитывать. Опишем необходимые для этого действия.

Поскольку одновременно потребуется несколько независимо действующих групп объектов, потребуем, чтобы для коррекции изменений T_i не требовалось какого-либо центрального устройства (лидера). Все группы O_i будут инициировать коррекцию, выполняя следующие действия.

Приемник посылает в RS_k синхросигнал S длительностью не менее T, если до этого момента он не обнаружил присутствия в RS_k сигнала S, посланного другими источниками. В результате суперпозиции сигналов S образуется совместный сигнал S переменной длительности. Момент его окончания ${}^*S$ служит сигналом начала синхронизации. Пусть приемники имеют порядковые номера. Приемник с наименьшим номером измеряет расстояние до RS_k. Затем остальные приемники измеряют его по очереди. Следующий приемник начинает измерение после известного объектам момента окончания измерения предыдущим приемником. Когда все объекты в группе завершены, процесс измерения повторяется. Измерения проводятся в соответствии с подразделом 1.2, следуя решениям [**6**–**10**]. После измерений времени синхронизация завершается.

Возможен более сложный метод измерения, позволяющий выполнять измерения для всех приемников за время T, описанный в работе [**14**]. Его принцип действия следующий. Выше в данном разделе объект корректирует расстояние от RS только после того, как все объекты завершили измерение расстояния. Новый метод использует выделенный только для него канал. Пусть сначала объекты выполняют

описанное выше поочередное измерение расстояний до RS. После этого объекты посылают сигнал S. Получение объектом сигнала ${}^*S$ служит началом ускоренной синхронизации. При получении сигнала ${}^*S$ с задержкой D_i каждый объект O_i передает в RS шкалу Q.

Шкала представляет собой последовательность двоичных разрядов. Каждый разряд шкалы приписывается одному из O_i, участвующих в синхронизации. Объект O_i помещает в свой разряд шкалы с порядковым номером i сигнал St. Этот сигнал имеет меньшую длительность, чем интервал времени, отведенный для передачи разряда шкалы. Сигнал размещается в центре разряда. Шкалы объектов с выравниванием одноименных разрядов поступают в RS и возвращаются объектам. При изменении расстояний между O_i и RS сигнал St смещается внутри разряда, позволяя всем объектам учитывать изменение расстояния до RS за время передачи шкалы.

Дополнение. Пусть для измерения расстояний используются сигналы с частотами, отличными от частот сигналов сообщений. Пусть также в RS используется дополнительный ретрорефлектор без модулятора (или модулятор всегда пропускает сигналы измерения). В этом случае измерения ускоряются, поскольку все объекты могут измерять расстояния одновременно. Этот результат следует из свойства ретрорефлектора возвращать сигнал, прошедший через направленный канал, к его источнику. Получено ускорение, без которого каждому объекту потребуется выделять временной интервал T для измерения расстояния до RS_k.

2.2. Многоуровневая синхронизация приемников команд

Разделим группу приемников на несколько подгрупп (слоев), как показанона рисунке 4. В пределах одного слоя объекты действуют так, как показано выше. Сигналы, которыми обмениваются объекты одного слоя, недоступны в других слоях. Повторители ${}^*RS_k$ соседних слоев доступны объектам этих слоев и могут быть объединены в одно устройство. Первый слой приемников получает команды источника от ретранслятора своего слоя. Затем один из приемников первого слоя действует через ретранслятор как источник для приемников второго слоя. Приемники последующих слоев действуют аналогично. В простейшем случае требуется, чтобы только приемники последнего слоя действовали синхронно с внешними объектами. Здесь внешний объект — это объект внешней среды или объект СК, не включенный в конкретный процесс синхронизации. В более сложном случае компьютеры объектов всех слоев будут одновременно выполнять совместные

действия с внешними объектами. В последнем случае возможны два варианта. В первом действия каждого из слоев завершаются в течение одного и того же известного промежутка времени. Во втором варианте время завершения действий слоев различны и заранее неизвестны. Ниже рассмотрена синхронизация внешних действий только последнего слоя (процесс 1) и одновременно всех слоев (процесс 2).

Рассмотрим детально оба процесса.

Процесс 1.

Имеется k слоев объектов, $k = 0, \ldots, n$. Объекты — приемники слоя k для взаимодействия с приемниками слоя $k + 1$ выделяют один источник, который синхронизирует приемники слоя $k + 1$. Процесс достигает последнего слоя, и его приемники воздействуют на внешние объекты одновременно или с заданными дополнительными сдвигами во времени.

Объекты промежуточных слоев не воздействуют на внешние объекты, включая объекты СК. Для их воздействия предназначен следующий процесс 2.

Процесс 2.

Пусть для завершения работы любого слоя требуется интервал времени C. После получения команды приемники произвольного слоя $k \leq n$ многократно вычисляют значение $F = n - k$, увеличивая k на 1. Каждое следующее ‘вычисление F выполняется после временной задержки C. Когда достигается $F = 0$, приемники слоя выполнят внешнее действие. Таким образом, первое вычисление для слоя $k = 0$ дает $F = n$, для слоя $k = 1$ дает $F = n - 1$, и так далее. В результате, при получении команды в последнем слое, приемники во всех слоях получают $F = 0$ и одновременно выполняют необходимые действия.

Если объекты действуют *асинхронно*, то оптимальное время выполнения действий достигается с помощью процесса синхронизации, состоящего из двух шагов.

Шаг 1: **Подготовка к синхронизации**. На этом шаге источники первого уровня асинхронно подготавливают данные для передачи их приемникам первого уровня. Все действия выполняются с использованием барьерной синхронизации [11]. В ней один или несколько объектов, выполняющих совместную операцию, при ее выполнении посылают сигнал B на ретранслятор, доступный всем объектам.

Когда работают несколько объектов, и какой-то объект завершает свою работу, этот объект прекращает передачу сигнала β. Передача сигнала B прекращается, когда все участники

завершают операцию. Его отсутствие позволяет другим объектам начать следующую операцию. Приемники первого слоя обычно асинхронно готовят дополнительные данные для предварительной подготовки действий источников второго слоя.

Переход к действиям второго слоя осуществляется по сигналу барьерной синхронизации, информирующего объекты о том, что все они завершили подготовку к синхронизации. Таким образом, все слои выполняются по очереди. Пока все слои не нуждаются в одновременном выполнении внешних действий. Они будут выполняться на следующем шаге.

Шаг 2: **Синхронизация**. Процесс аналогичен процессу 2, но все слои перенумеровываются в обратном порядке. Последний слой n теперь имеет номер 0. Для синхронизации объекты слоя с новой нумерацией $k = 0$ выступают в качестве источников и начинают синхронизировать приемники всех слоев.

После получения команды объекты многократно вычисляют значение ${}^*F = n - k$, увеличивая k на 1 с задержкой после каждого вычисления ${}^*C$. Когда $F = 0$ получено во всех слоях, приемники одновременно выполняют внешнее действие.

Здесь синхронизация происходит быстрее, так как объекты не выполняют никаких действий, кроме отправки команды в предыдущий слой.

Таким образом, без использования общего центра управления приемники синхронизируют свои действия и выполняют их одновременно во всех слоях рассматриваемой многослойной объектной структуры.

3. Синхронизация действий источников команд

3.1. Синхронизация процессов обмена сообщениями

Синхронные действия группы источников подробно рассмотрены в [14], здесь мы дадим краткое и более близкое к тексту раздела 2 резюме синхронных действий группы источников. Источники действуют следующим образом, аналогично действиям приемников в разделе 2. Источники So_j из группы источников So, упорядоченных по j, посылают в RS сигнал S.

В ответ So_j получает от RS сигнал Srs и сигнал ${}^*Srs$ — признак завершения S. После этого So_j поочередно определяют расстояние до RS и могут вычислить задержки $D_j = 2(T - T_j)$, где T, T_j соответствуют T, T_i в разделе 2.

Теперь источники могут посылать синхронные сообщения (и команды приемникам) в RS без использования выделенного центра.

Для этого источники используют логическую шкалу LS — последовательность двоичных битов, равную числу источников в So. Источник So_j, который должен послать сообщение в RS, ставит в соответствующий ему бит j шкалы LS единицу и передает в RS сигнал с несущей частотой f_1. Остальные биты LS могут не содержать сигналов, либо So_j вносит ноль, который передается сигналом f_0.

Источники для начала работы с RS, используя шкалы, посылают в RS сигнал S и получают в ответ Srs и ${}^*Srs$. Затем, используя задержки D_j, источники посылают свои шкалы LS в RS, чтобы получить шкалу ${}^*LS$, которая объединяет одноименные разряды всех шкал, полученных в RS. Теперь So_j могут отправлять свои сообщения в RS упорядоченно, не задерживаясь на источниках, которые не запросили передачу сообщения.

RS теперь действует как единый источник, посылая сообщения или команды от RS к приемникам.

При асинхронных действиях объектов отсутствует возможность указать время занятия RS объектом. Необходимо применять барьерную синхронизацию (см. подраздел 2.2). Без нее значение C придется выбирать излишне большим.

Таким образом, для передачи своих сообщений объекты должны выполнить следующие действия. Объекты децентрализовано посылают сигналы S в RS, получают от RS сигнал ${}^*Srs$, по очереди определяют расстояние до RS с использованием задержек D_j, посылают в RS свои шкалы LS, получают от RS совместную шкалу, и получают право упорядоченно передавать сообщения.

3.2. Изменение порядка передачи сообщений источников

В подразделе 3.1 шкала представляет собой фиксированную конструкцию, в которой количество двоичных разрядов соответствует количеству источников. Каждый бит шкалы постоянно закреплен за определенным источником, номер которого совпадает с порядковым номером бита в шкале. Однако в СК при решении задачи может потребоваться как можно быстрее изменить этот фиксированный шкалой порядок. Ниже приведен вариант такого изменения.

Во-первых, каждый источник, требующий срочной передачи, должен сформировать код приоритета обслуживания. Он состоит из порядкового номера источника, которому предшествует группа двоичных разрядов приоритета. Двоичное число, представленное этой группой, оценивает приоритет в обслуживании сообщения. Чем выше это число, тем выше приоритет обслуживания. Затем источники посылают

команду уведомления, чтобы начать формирование новой шкалы с изменением порядка сообщений. После этих действий источники синхронно выполняют следующий[1] процесс 1 для определения объекта с текущим наивысшим приоритетом.

Процесс 1

Шаг 1: Источник передает в RS значение старшего двоичного разряда своего кода приоритета (старшего из разрядов, не переданных в этом процессе ранее). Значение «1» передается сигналом частоты f_1, значение «0» — сигналом частоты f_0.

Шаг 2: Если источник на шаге 1 послал сигнал f_0 и получил сигнал f_1 от других источников, он завершает процесс 1. Остальные источники переходят к шагу 3.

Шаг 3: Источник проверяет, не остались ли не переданными какие-либо биты кода приоритета на шаге 1. Если таковые имеются, источник возвращается к шагу 1. В противном случае процесс 1 завершен.

Этот процесс отделяет один источник от претендентов на изменение последовательности обслуживания. Если окажется, что несколько источников имеют одинаковые приоритеты, то затем порядковые номера источников выделят только один из них.

Далее выполняется процесс 2 для формирования шкалы доступа RS с учетом приоритетов.

Процесс 2

- Начинается процесс формирования новой шкалы доступа к RS. В начале шкалы вводится дополнительная зона приоритетов, состоящая из одного бита. Изначально эта зона содержит ноль и игнорируется источниками. Для передачи сообщения раньше всех других источников источник должен ввести единицу в зону и ввести ноль в свой разряд шкалы.
- Процесс запроса приоритетной передачи завершен.
- Теперь источник с единицей в зоне передает сообщение, за ним следуют другие источники, которые поместили единицу в свой разряд шкалы. После отмены срочной передачи сообщения источник в зоне должен убрать единицу.

[1]Исходным вариантом процесса 1 является метод ДПУ (децентрализованное управление приоритетами), при котором право на передачу сообщения получает объект с наивысшим текущим приоритетом. Для проводной шины ДПУ был разработан в ИАТ (позднее ИПУ) АН СССР в 1970 году [15]. ДПУ использовался в промышленных системах управления. Позже аналогичное решение было предложено в компании Philips и широко используется для разрешения конфликтов в качестве интерфейса I^2C.

Существенно, что в разделе 2 приемники в нескольких случаях выступают в роли источников. Поэтому все вышесказанное относится и к ним.

Таким образом, получен оперативный учет требований при изменении прав объектов на передачу сообщений.

4. Особенности группового взаимодействия объектов

В этом разделе выделены основные действия, выполняемые в статье групповыми операциями, и приведены некоторые примеры других групповых операций, расширяющих возможности распределенных систем. Приведенные ниже операции были разработаны в Институте проблем управления РАН; некоторые из них использовались в промышленных системах управления.

4.1. Краткое описание групповых управляющих операций, используемых в статье

В статье групповые операции используются для управления действиями объектов. При этой операции одноименные биты сообщений объектов поступают в RS одновременно и обрабатываются без задержки и без использования вычислительных средств. Операции выполняют следующие действия.

1. Начальный перевод асинхронно работающих объектов в синхронное состояние с помощью сигналов S и ${}^*S$ в подразделе 2.1. Операция выполняется с использованием или без использования RS (см. ниже).
2. Формирование задержек в передаче сообщений объектами в RS для одновременной доставки в RS.
3. Коррекция приемниками времени использования сообщений, полученных от RS в разное время, в зависимости от удаленности приемников от RS. Коррекция позволяет приемникам выполнять свои действия одновременно или с установленными дополнительными временными задержками.
4. Варианты многоуровневой синхронизации объектов. Предусматривается синхронное выполнение действий объектами последнего слоя, промежуточных слоев, итеративная коррекция действий объектов предыдущих слоев с учетом результатов, полученных в последующих слоях.
5. Применение шкал для ускорения измерения расстояния между объектами; одновременное разрешение конфликта доступа для группы объектов; изменение порядка доступа с учетом текущего приоритета доступа объекта.

6. Применение барьерной синхронизации для синхронизации группы объектов, каждый из которых выполняет общую задачу для группы, действуя асинхронно.

4.2. Вычислительные групповые операции, выполняемые на ретрансляторе

Следующие операции включают распределенные вычислительные операции, которые ускоряют управление системой при поиске объектов с заданным набором свойств и оценке системы в целом. Это побитовые операции ***И*** и ***ИЛИ***, нахождение **max** и **min** чисел, аналого-цифровые арифметические операции. В данной работе такие операции не используются. Однако они включены в раздел, так как позволяют оценить состояние всех объектов и данных в них одновременно за время, которое не зависит от количества участников операции.

Операции побитового ***И*** и ***ИЛИ*** позволяют быстро оценить состояние всех объектов в системе. Для этого состояние объекта описывается шкалой — последовательностью двоичных разрядов. Каждый из них равен единице при наличии соответствующего признака и равен нулю при его отсутствии и передается соответственно сигналами с частотами f_1 и f_0.

Оценка состояния всех объектов производится при одновременной передаче шкал в RS с совмещением в RS одноименных разрядов шкал объектов. При выполнении операции ***И*** наличие в RS разряда шкалы f_0 означает отсутствие соответствующего признака хотя бы в одном объекте. В противном случае признак присутствует во всех объектах. Для операции ***ИЛИ*** в тех же условиях наличие f_1 означает, что, хотя бы в одном объекте признак присутствует. В противном случае все объекты не имеют соответствующего признака.

При определении максимального значения чисел используется логическая шкала с разрядами, представленными в произвольной системе счислении. Все разряды в шкале содержат ноль, за исключением разряда, соответствующего значению цифры. Объекты посылают цифры числа в RS для определения операции **max**. Числа должны поступать в RS синхронно с совпадением одноименных цифр.

На первом этапе объекты передают старшую цифру числа. В передаче следующей цифры участвуют только те объекты, которые до этого передали наибольшую из цифр. Затем в передаче следующей цифры участвуют только те объекты, которые ранее передали наибольшую из цифр, и процесс повторяется. В результате остается максимальное из переданных чисел. С заменой единиц на нули определяется **min**.

В качестве примера используем десятичную систему. С помощью шкал цифры 7, 3 и 1 записываются как 001000000, 000000100, 000000001. При объединении одноименных разрядов этих шкал в RS, все объекты получат результат объединенной новой шкалы 001000101. Из нее следует, что **max** равно 7, а **min** равно 1.

Результат будет получен за время, не зависящее от количества объектов, участвующих в групповой операции. Для распределенных систем полезно увеличить основание системы счисления, что уменьшает количество передач по каналам связи.

Рассмотрим выполнение аналого-цифровых операций. Для выполнения аналого-цифровых операций RS должен содержать аналого-цифровой преобразователь (АЦП). Покажем выполнение операции сложения чисел. Энергия оптических сигналов, поступающих в RS с совпадением по времени, измеряется фотодетектором, который передает результат измерения на АЦП. Последний преобразует результат в число и рассылает его всем объектам.

Как и в предыдущем примере, числа, передаваемые объектами в RS, представляются в десятичной системе счисления с помощью шкал. Пусть три объекта передают в RS комбинацию цифр шкал трех чисел 789, 988, 786, соответственно. При применении шкал эти числа будут записаны как

$$[001000000; 010000000; 100000000],$$
$$[100000000; 010000000; 010000000],$$
$$[001000000; 010000000; 000100000].$$

При объединении цифр шкалы в RS объекты получат три шкалы из RS

$$[102000000; 030000000; 110100000].$$

Здесь цифры 3 и 2 в первой и второй шкалах показывают цифровые показания АЦП, суммирующие энергию трех и двух сигналов. В результате каждый объект, используя свое вычислительное средство, локально выполняет суммирование и получает результат

$$(9 + 2 \times 7) \times 100 + 3 \times 8 \times 10 + 9 + 8 + 6 = 2300 + 240 + 23 = 2563.$$

Таким образом, вычисление суммы не зависит от количества объектов, участвующих в операции сложения. Для вычитания достаточно получить две суммы в RS для уменьшаемого и вычитаемого чисел и завершить операцию локально в каждом объекте.

В частности, сложение может быть использовано для операции счет. Например, при поиске **max** по энергии поступающих сигналов определяется количество одинаковых чисел **max** в объектах.

Для получения в СК точного цифрового значения аналого-цифровых операций при суммировании нескольких тысяч сигналов необходимы источники со стабильным значением энергии оптического сигнала. В [16] приведен простой светодиодный источник со стабильностью выходной мощности лучше 50 ppm/°C.

4.3. Три основных механизма ускорения синхронизации объектов

Из предыдущих разделов ясно, что в синхронизации действий объектов центральную роль играют три механизма. К ним относятся начальная синхронизация асинхронных объектов сигналами S и ${}^*S$, замена группы источников сигналов одним повторителем сигналов RS и логические шкалы, представляющие ноль и единицу активными сигналами.

Хотя в подразделе 2.1 объекты используют RS при начальном запуске, его использование необходимо только для получения высоких скоростей взаимодействия объектов. Например, объекты могут не использовать RS для связи с периферией. В этом случае объекты могут непосредственно получать сигналы S и определять наличие ${}^*S$ без RS. Объекты, получившие ${}^*S$, будут поочередно выполнять необходимые операции.

Однако эти операции будут выполняться значительно медленнее, чем при использовании RS. Без RS после появления в системе сигнала ${}^*S$ первый объект обнаружит ${}^*S$ и начнет передачу сообщения не позднее момента времени T. Следующий объект начнет передачу не позднее момента времени T после детектирования передачи предшественника и так далее. Между сообщениями возникает неуправляемая пауза длительностью $\leqslant T$. Добавление RS устранило зависимость от T.

Отметим важность RS для групповых операций. Эти операции требуют, чтобы все объекты получили каждый бит всех одновременно передаваемых сообщений без помех от других битов. Поэтому без RS каждый бит групповой операции находится в тех же условиях, что и все сообщение выше. Это приводит к значительному замедлению групповых операций.

Логические шкалы устраняют конфликт доступа объектов к RS одновременно для всех конфликтующих объектов. Их второе важное применение – создание групповых аналого-цифровых вычислительных операций.

Заключение

Предложенный в статье метод предоставляет следующие возможности. Метод ускоряет начало выполнения одновременных совместных действий группы распределенных устройств (объектов) суперкомпьютера в ответ на неожиданно обнаруженные объектами события. Объекты синхронизируют свои действия децентрализовано, взаимодействуя друг с другом посылкой групповых команд, без участия центра синхронизации. Групповые операции одновременно обрабатывают сообщения многих распределенных объектов во время передачи группы сообщений, не замедляя передачу. Длительность выполнения групповой операции не зависит от количества ее участников.

Специальное представление передаваемых между объектами данных позволяет групповым операциям выполнять ряд видов распределенных вычислений непосредственно в простых средствах сети без привлечения компьютеров.

Все приведенные в статье операции не требуют вмешательства в структуру и могут выполняться во внешних устройствах, подобных сетевым картам компьютеров.

Список литературы

[1] Tennenhouse D., Smith J. M., Sincoskie . D., et al *A Survey of Active Network Research* // IEEE Communications Magazine.– January 1997.– Vol. **35**.– No. 1.– pp. 80-86. ↑5

[2] Zilberman N., Watts P. M., Rotsos C., Moore A.W *Reconfigurable Network Systems and Software Defined Networking* // *Proc. of the IEEE*.– vol. **103**.– 2015.– pp. 1102 – 1124. ↑5

[3] Tokusashi Y., Dang H. Tu, Pedone F., Soulé R., Zilberman N he Case For In-Network Computing On Demand // *Proceedings of the Fourteenth EuroSys Conference*, EuroSys '19.– March 2019.– pp. 1-16. ↑6

[4] Sapio A, Abdelaziz I, Aldilaijan A, et al *In-network computation is a dumb idea whose time has come* // *Proceedings of the 16th ACM Workshop on Hot Topics in Networks*.– 2017.– pp. 150–156. ↑6

[5] Daehyeok Kim *Towards Elastic and Resilient In-Network Computing.* ↑6

[6] *IEEE Std 1588-2008 (Revision of IEEE Std 1588-2002)*.– 24 July 2008,. ↑6, 12

[7] *IEEE 1588-2019 — IEEE Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems. IEEE Instrumentation and Measurement Society.*.– 2020. ↑6, 12

[8] F. Girela-López, J. López-Jiménez, M. Jiménez-López, R. Rodríguez, Ros E., Díaz J *IEEE 1588 High Accuracy Default Profile: Applications and Challenges* // IEEE Access.– 2020.– Vol. **8.**.– pp. 45211–45220. ↑6, 12

[9] Sliwczynski Ł., Krehlik P., Buczek Ł., Schnatz H. *Picoseconds-Accurate Fiber-Optic Time Transfer With Relative Stabilization of Lasers Wavelengths* // Journal of lightwave technology.– september 15, 2020.– Vol. **38,**.– No. 18,.– pp. 5056–5063. ↑6, 12

[10] Moreira P., Darwazeh I *Digital femtosecond time difference circuit for CERN's timing system.* ↑6, 11, 12

[11] Stetsyura G *Addition for Supercomputer Functionality* // *Separate volume of the Communications in Computer and Information Science (CCIS) series with subtitle "Supercomputing".*– vol. **687**, M: Springer International Publishing AG.– 2017.– pp. 251–263. ↑6, 14

[12] Rabinovich W. S., Goetz P. G., Mahon R. et al. *45-Mbit/s cat's-eye modulating retroreflectors* // Optical Engineering.– 2007.– Vol. **46**.– No. 10.– pp. 1-8. ↑8

[13] Zhu Y., Wang G *Research on Retro-reflecting Modulation in Space Optical Communication System* // *IOP Conference Series Earth and Environmental Science.*– January 2018. ↑8

[14] Стецюра Г. Г. *Децентрализованная автономная синхронизация процессов взаимодействия мобильных объектов* // Проблемы управления.– 2020.– № 6..– с. 47–56. ↑12, 15

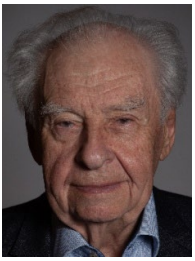
[15] Stetsura G. G. *Comments on "Peripheral Interface Standards for Microprocessors"* // *Proceedings of the IEEE.–1977..*– vol. **65**.– pp. 1920. ↑17

[16] Bosiljevac M., Babić D., Sipus Z. *Temperature-Stable LED-Based Light Source without Temperature Control* // *Proceedings of SPIE OPTO.*– vol. **9754**, San Francisco, CA, USA,.– 2016.– pp. 1–6.. ↑21

Поступила в редакцию 01.07.2022;
одобрена после рецензирования 18.09.2022;
принята к публикации 19.09.2022.

Рекомендовал к публикации *д.ф.-м.н. С. М. Абрамов*

Информация об авторе:



Геннадий Георгиевич Стецюра

д. т. н., профессор, гл. н. с. ИПУ РАН. Область интересов: информатика, computing. Основные работы в области распределенных многокомпонентных цифровых систем. Более 150 публикаций и более 20 патентов и авторских свидетельств на изобретения.

0000-0003-4606-4424
e-mail: gstetsura@mail.ru

Автор заявляет об отсутствии конфликта интересов.

ISSN 2079-3316 **PROGRAM SYSTEMS: THEORY AND APPLICATIONS** **vol. 13**, No 4(55), pp. 3–26
Research Article HARDWARE AND SOFTWARE FOR SUPERCOMPUTERS

UDC 004.272.45
10.25209/2079-3316-2022-13-4-3-26;

Synchronous execution of group operations in distributed supercomputer components and computer clusters

Gennady Georgievich **Stetsyura**
ICS V. A. Trapeznikov of RAS

(*learn more about the author in Russian on p. 23*)

Abstract. This paper proposes decentralized processes for synchronizing the actions of a distributed group of active components (objects) in supercomputers and computer clusters, allowing them to move to specified states or influence the external environment synchronously. The object action depends on the current state of the object and the external environment. The actions should start with the minimum delay after the possibility of their execution is detected. Synchronization is performed by exchanging optical signals over wireless communication channels through an optical signal repeater, combining one group of objects or sequences of groups of objects (layers). Accurate distance measurement performs the compensation of possible changes in distances between objects. Group operations accelerate synchronize and simultaneously receive data from a group of distributed objects. Data processing occurs during their transfer, without increasing the time. The operation time does not depend on the quantity of data processed by the operation. A group operation is performed in a repeater containing no computational means. (*In Russian*).

Key words and phrases: supercomputers, group operations, decentralized control, multilayer synchronization of object actions, distributed in-network computing

2020 *Mathematics Subject Classification*: 65Y05; 68Q10

For citation: Stetsyura G. G. *Synchronous execution of group operations in distributed supercomputer components and computer clusters* // Program Systems: Theory and Applications, 2022, **13**:4(55), pp. 3–26. (*In Russian*). http://psta.psiras.ru/read/psta2022_4_3-26.pdf

References

[1] Tennenhouse D., Smith J. M., Sincoskie . D., et al. “A Survey of Active Network Research”, *IEEE Communications Magazine*, **35**:1 (January 1997), pp. 80-86.↑5

[2] Zilberman N., Watts P. M., Rotsos C., Moore A.W. “Reconfigurable

Network Systems and Software Defined Networking", *Proc. of the IEEE*, vol. **103**, 2015, pp. 1102 – 1124.↑5

[3] Tokusashi Y., Dang H. Tu, Pedone F., Soulé R., Zilberman N he Case For In-Network Computing On Demand, *Proceedings of the Fourteenth EuroSys Conference*, EuroSys '19, March 2019, pp. 1-16. ↑6

[4] Sapio A, Abdelaziz I, Aldilaijan A, et al. "In-network computation is a dumb idea whose time has come", *Proceedings of the 16th ACM Workshop on Hot Topics in Networks*, 2017, pp. 150–156. ↑6

[5] Daehyeok Kim. *Towards Elastic and Resilient In-Network Computing.* ↑6

[6] *IEEE Std 1588-2008 (Revision of IEEE Std 1588-2002)*, 24 July 2008,. ↑6, 12

[7] *IEEE 1588-2019 — IEEE Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems. IEEE Instrumentation and Measurement Society.*, 2020. ↑6, 12

[8] F. Girela-López, J. López-Jiménez, M. Jiménez-López, R. Rodríguez, Ros E., Díaz J. "IEEE 1588 High Accuracy Default Profile: Applications and Challenges", *IEEE Access*, **8.** (2020), pp. 45211–45220.↑6, 12

[9] Sliwczynski Ł., Krehlik P., Buczek Ł., Schnatz H.. "Picoseconds-Accurate Fiber-Optic Time Transfer With Relative Stabilization of Lasers Wavelengths", *Journal of lightwave technology*, **38,**:18, (september 15, 2020), pp. 5056–5063.↑6, 12

[10] Moreira P., Darwazeh I. *Digital femtosecond time difference circuit for CERN's timing system.* ↑6, 11, 12

[11] Stetsyura G. "Addition for Supercomputer Functionality", *Separate volume of the Communications in Computer and Information Science (CCIS) series with subtitle "Supercomputing"*, vol. **687**, Springer International Publishing AG, M, 2017, pp. 251–263.↑6, 14

[12] Rabinovich W. S., Goetz P. G., Mahon R. et al.. "45-Mbit/s cat's-eye modulating retroreflectors", *Optical Engineering*, **46**:10 (2007), pp. 1-8.↑8

[13] Zhu Y., Wang G. "Research on Retro-reflecting Modulation in Space Optical Communication System", *IOP Conference Series Earth and Environmental Science*, January 2018. ↑8

[14] Stetsura G. G.. "Decentralized autonomic synchronization of interaction processes of mobile objects", *Problemy upravleniya*, 2020, no. 6., pp. 47–56 (in Russian). ↑12, 15

[15] Stetsura G. G.. "Comments on "Peripheral Interface Standards for Microprocessors"", *Proceedings of the IEEE.–1977.*, vol. **65**, pp. 1920.↑17

[16] Bosiljevac M., Babić D., Sipus Z.. "Temperature-Stable LED-Based

Light Source without Temperature Control", *Proceedings of SPIE OPTO*, vol. **9754**, San Francisco, CA, USA,, 2016, pp. 1–6.. ↑21