

Алгоритмы индексирования для памяти РСМ

Рафикова Э.Р., elonarafikova@gmail.com

Аннотация

В данной работе были исследованы алгоритмы хеширования для памяти типа РСМ, который является относительно новым видом памяти, в частности перспективным видом основной памяти. Для этого были изучены его характеристики и свойства, было произведено сравнение РСМ с другими типами памяти.

Основным результатом работы является модифицированный алгоритм расширяемого хеширования РСМFЕН (PCM Friendly Extensible Hashing). Он представляет собой адаптацию стандартного расширяемого хеширования для памяти с изменением фазового состояния.

Введение

Самые разные вычислительные системы немислимы без использования устройств хранения. Но до сих пор нет универсального типа памяти, который удовлетворял бы всем критериям. Большая разница в производительности между запоминающими устройствами, а также все новые и новые требования от электронных систем, например портативных электронных устройств, открывают новые возможности для устройств хранения.

Память с изменением фазового состояния (Phase Change Memory, сокращенно РСМ) - одно из многообещающих устройств хранения. Это новый физический принцип хранения информации, который работает посредством изменения фазового состояния халькогенидного стекла [1].

В этой работе предложен модифицированный метод расширяемого хеширования [2], который является дружественным к РСМ, то есть является более эффективным для памяти с изменением фазового состояния, чем обычный. Также был проведен ряд экспериментов, доказывающих этот факт.

Запись в РСМ

Одной из основных проблем эффективного использования РСМ являются ограничения, связанные с записью данных в память.

По сравнению с чтением ячейки РСМ, операция записи потребляет более высокий ток, использует более высокое напряжение и занимает больше

времени из-за физических свойств памяти. Получается, что операция записи потребляет в 6-10 раз больше энергии, чем чтение.

В РСМ устройстве задержка записи в ячейке примерно в три раза больше, чем при операции чтения. Более того, многие прототипы такого вида памяти поддерживают итеративную запись, то есть ограниченное число битов на одну итерацию, для того чтобы сократить мгновенный уровень тока. В будущем это явление скорее всего сохранится, особенно для систем с небольшими физическими ресурсами. Тогда из-за ограниченной пропускной способности получается еще большая задержка записи.

Существующие прототипы предоставляют возможность делать ограниченное количество перезаписи в ячейку, а именно от 10^6 до 10^8 операций записи на ячейку. Получается, что РСМ, даже при хороших алгоритмах выравнивания износа, как основная память, может сохраняться только в течение нескольких лет при реальных нагрузках.

Исходя из вышесказанного, основной задачей при разработке алгоритмов, дружественных к РСМ, является преодоление асимметрии между РСМ-чтением и РСМ-записью. Поэтому одной из целей разработки эффективных алгоритмов для РСМ является минимизация количества операций записи.

PCMFEN

Авторы существующих решений на тему применимости разного вида хеш-таблиц для памяти типа РСМ не рассматривали алгоритм расширяемого хеширования.

Расширяемое хеширование часто используется в базах данных, так как базы данных могут быть крайне большими и перехеширование всей базы данных займет продолжительное время, при этом лишая пользователей доступа к ней. А при использовании этого алгоритма перехешировать придется только малые группы, что не сильно замедлит работу базы данных.

Далее рассматривается и описывается модификация алгоритма расширяемого хеширования, предложенная в этой работе (PCMFEN - PCM friendly extensible hashing). Эта модификация учитывает особенности памяти с изменением фазового состояния.

Метод расширяемого хеширования заключается в том, что хеш-таблица представлена как каталог, каждая ячейка которого указывает на страницы в нем. А также имеется определенный алгоритм вставки пары "ключ-значение". Сама хеш-таблица будет иметь глобальную глубину, а каждая из страниц имеет локальную глубину. Глобальная глубина показывает сколько последних бит будут использоваться для хеш-функции. А из разницы локальной глуби-

ны и глобальной глубины можно понять сколько ячеек каталога ссылаются на страницу. Это можно показать формулой $K = 2^{(G-L)}$ где G - глобальная глубина, L - локальная глубина, а K - количество ссылающихся ячеек.

В основу измененного алгоритма легла идея о допустимости переполнения. А именно: структура хранения данных модифицируется так, что страница может иметь на одну или несколько пар "ключ-значение" больше, чем позволяет максимальный размер страницы.

Для реализации такой идеи введем два новых понятия:

1. Коэффициент переполнения - это максимальное количество пар, которые можно вставить в уже полностью заполненную страницу. (обозначим OVF)
2. Глубина переполнения страницы - это разность между максимальным размером страницы и фактическим. (Показывает, на сколько пар переполнена страница)

Тогда модифицированный алгоритм добавления пары такой:

1. Переводим ключ в двоичный вид, смотрим на последние G битов и решаем, в какую страницу поместить значение.
2. Если страница имеет свободное место, то помещаем туда значение, если же страница, куда следует положить значение переполнена, то смотрим на глубину переполнения:
 - (a) Если она меньше, чем коэффициент переполнения, значит в данную страницу можно еще поместить значение, помещаем его и увеличиваем глубину переполнения на 1.
 - (b) Если же глубина переполнения равна коэффициенту переполнения, то значит следует посмотреть на локальную глубину:
 - i. Если она меньше, чем глобальная глубина, то значит на страницу есть несколько указателей и нам достаточно переиндексировать ее, разделив при этом на две и занести значения в новые две страницы, увеличив их локальную глубину на 1, и изменить значение глубины переполнения на разность между суммой максимального размера страницы с коэффициентом переполнения и фактическим размером страницы.
 - ii. Если же локальная глубина была равна глобальной, то мы увеличиваем глобальную глубину на 1, удваивая при этом количество ячеек в каталоге, количество указателей на страницы, а также увеличиваем количество последних бит, по которым

мы распределяем значения. Далее локальная глубина переполненной страницы становится меньше, чем глобальная глубина и мы выполняем предыдущий шаг, то есть перехешируем нужную страницу, разделим ее на две страницы и так далее.

Сравнение расширяемого хеширования и РСМФЕН

Начнем с того, что сравним количество операций при построении структуры. В модифицированном методе количество операций должно вырасти за счет добавлений коэффициента переполнения и глубины переполнения.

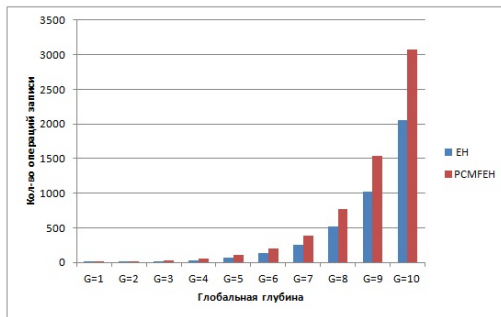


Рис. 1: Сравнение алгоритма расширяемого хеширования (EH) и модифицированного метода (РСМФЕН). Зависимость количества операций записи при построении структуры от глобальной глубины (G)

Как и предполагалось, у РСМФЕН количество операций записи больше, чем у обычного расширяемого хеширования, особенно эта разница заметна при глобальной глубине начиная с 8. Но, так как использования большого значения глобальной глубины целесообразно при большом количестве вставляемых пар, даже такая разница не повлияет на производительность модифицированного метода, так как это разница будет несущественна по сравнению с общим количеством операций записи. Этот факт будет далее подтвержден экспериментами.

Теперь будем вставлять по 1000 пар, в которых ключи и значения будут числами от 0 до 100 000, значение глобальной глубины рассмотрим при $G=2$ и $G=4$, размер страницы будем изменять от $BS=2$ до $BS=16$.

Показанные эксперименты подтверждают теоретическое предположение о сокращении количества операций записи в алгоритме РСМФЕН. Уменьшение происходит при любой глубине и размере страницы.

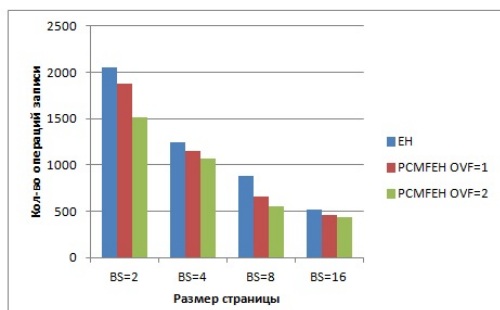


Рис. 2: Сравнение расширяемого хеширования (EH) с PCMFEN с коэффициентами переполнения 1 и 2 (Глобальная глубина $G=2$)

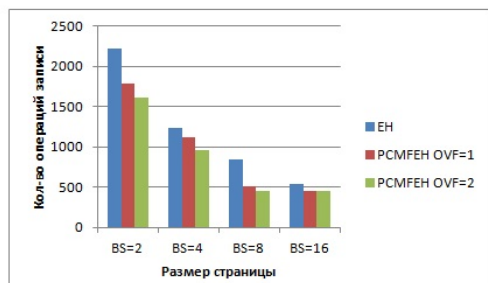


Рис. 3: Сравнение расширяемого хеширования (EH) с PCMFEN с коэффициентами переполнения 1 и 2 (При глобальной глубине $G=4$)

Теперь проверим нашу модификацию на большом объеме данных, а именно будем заполнять структуру 100 000 пар "ключ-значение", значение глобальной глубины и размер страниц будем менять. На Рис. 4 показаны результаты для стандартного расширяемого хеширования и PCMFEN.

Как мы видели ранее, при глобальной глубине больше 8 в модифицированном методе совершается больше операций записи при построении структуры, но это количество не повлияло на результат, что подтвердилось проведенными экспериментами. Модифицированный метод показывает себя лучше стандартного алгоритма расширяемого хеширования и при вставке большого количества пар. (В частности, при большой глобальной глубине)

Получаем, что предложенный метод PCMFEN учитывает асимметрию между операциями чтения и записи данных в РСМ. А именно сокращает количество операций записи при вставке. Но кроме того, как было сказано, РСМ имеет ограниченное число перезаписей ячейки. Таким образом для эффективности важно не только уменьшение общего количества операций записи,



Рис. 4: Сравнение количества операций записи расширяемого хеширования (EH) и PCMFEN

но и количество перезаписи в ячейке по отдельности.

Для проверки, что и по свойству изнашиваемости PCMFEN лучше, чем стандартное хеширование, будем учитывать операции записи для каждой переменной, затем будем производить вставки пар, как в эксперименте выше, и среди полученных значений для каждой переменной возьмем максимум. Именно по этому значению будем сравнивать.

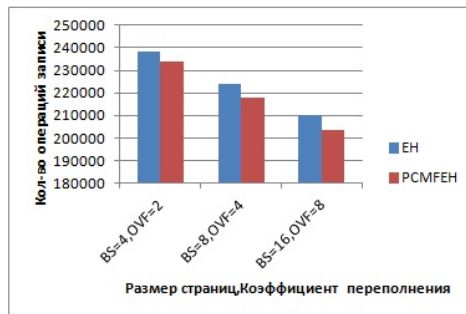


Рис. 5: Сравнение обычного и модифицированного метода по описанному выше параметру

Эксперименты доказали, что модифицированный метод эффективнее, чем стандартное расширяемое хеширование и по этому параметру, влияющему на свойство памяти изнашиваться.

Заключение

В данной работе были исследованы алгоритмы хеширования для памяти типа РСМ, который является относительно новым видом памяти, в частности перспективным видом основной памяти. Для этого были изучены его характеристики и свойства, было произведено сравнение РСМ с другими типами памяти.

Основным результатом работы является модифицированный алгоритм расширяемого хеширования (РСМФЕН). Он представляет собой адаптацию стандартного расширяемого хеширования для памяти с изменением фазового состояния.

Основная идея модификации состоит в разрешении переполнения страницы для уменьшения количества разделений страниц, и как следствие количества перестроений, которые приводят к многочисленным операциям записи. Были произведены эксперименты, показывающие, что измененный алгоритм учитывает проблемы, связанные с операцией записи в памяти РСМ, с помощью уменьшения общего количества операций записи в память. Также снижается количество операций записи на ячейки по отдельности, что положительно влияет на сохранность памяти РСМ, которая имеет ограниченное количество перезаписи в ячейку.

Литература

- [1] Chen Shimin, Gibbons Phillip B., Nath Suman. Rethinking Database Algorithms for Phase Change Memory.— 2011.—January.
- [2] Extendible Hashing - a Fast Access Method for Dynamic Files / Ronald Fagin, Jurg Nievergelt, Nicholas Pippenger, H. Raymond Strong // ACM Trans. Database Syst.— 1979.—sep.— Vol. 4, no. 3.— P. 315–344.