



(19) RU (11) 2 236 751 (13) C2  
(51) МПК<sup>7</sup> Н 03 М 7/30, 13/39

РОССИЙСКОЕ АГЕНТСТВО  
ПО ПАТЕНТАМ И ТОВАРНЫМ ЗНАКАМ

(12) ОПИСАНИЕ ИЗОБРЕТЕНИЯ К ПАТЕНТУ РОССИЙСКОЙ ФЕДЕРАЦИИ

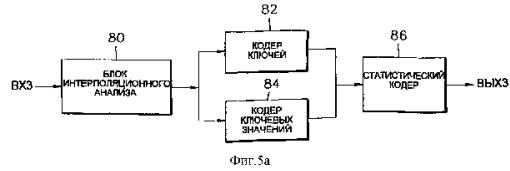
(21), (22) Заявка: 2001131591/09, 22.11.2001  
(24) Дата начала действия патента:  
22.11.2001 i.i. 1-19  
(30) Приоритет: 23.11.2000 KR 2000-70090 i. 20  
07.07.2001 KR 2001-40704  
(43) Дата публикации заявки: 10.08.2003  
(46) Дата публикации: 20.09.2004  
(56) Ссылки: RU 2140668 С1, 27.10.1999. US  
6028608 A, 22.02.2000. US 5801776 A,  
01.09.1998. US 6034697 A, 07.03.2000.  
(98) Адрес для переписки:  
129010, Москва, ул. Б. Спасская, 25, стр.3,  
ООО "Юридическая фирма Городисский и  
Партнеры", пат.пов. Ю.Д.Кузнецову, рег.№ 595

(72) Изобретатель: ДЗАНГ Еуе-сеон (KR),  
КИМ До-киоон (KR), ВОО Санг-оак (KR)  
(73) Патентообладатель:  
САМСУНГ ЭЛЕКТРОНИКС КО., ЛТД (KR)  
(74) Патентный поверенный:  
Кузнецов Юрий Дмитриевич

(54) СПОСОБЫ И УСТРОЙСТВО ДЛЯ СЖАТИЯ И ВОССТАНОВЛЕНИЯ ТРАЕКТОРИИ АНИМАЦИИ С  
ИСПОЛЬЗОВАНИЕМ ЛИНЕЙНОЙ АППРОКСИМАЦИИ

(57)  
Изобретение относится к анимации трехмерных графических моделей. Технический результат заключается в эффективном сжатии данных в форме интерполятора для обеспечения быстрой передачи и запоминания данных. Устройство сжатия включает блок интерполяционного анализа для выделения определенного количества точек разрыва из траектории анимации и вывода ключей и ключевых значений, соответствующих точкам разрыва,

кодер для кодирования ключей, кодер ключевых значений и статистический кодер. При осуществлении анализа точек разрыва получают закодированный поток битов, имеющий минимальное количество точек разрыва. 5 с. и 15 з.п. ф.-лы, 14 ил., 2 табл.



R  
U  
2  
2  
3  
6  
7  
5  
1  
C  
2

? 2 3 6 7 5 1 C 2



(19) RU (11) 2 236 751 (13) C2  
(51) Int. Cl. <sup>7</sup> H 03 M 7/30, 13/39

RUSSIAN AGENCY  
FOR PATENTS AND TRADEMARKS

(12) ABSTRACT OF INVENTION

(21), (22) Application: 2001131591/09, 22.11.2001

(24) Effective date for property rights:  
22.11.2001c1.1-19

(30) Priority: 23.11.2000 KR 2000-70090c1.20  
07.07.2001 KR 2001-40704

(43) Application published: 10.08.2003

(46) Date of publication: 20.09.2004

(98) Mail address:  
129010, Moskva, ul. B. Spasskaja, 25, str.3,  
OOO "Juridicheskaja firma Gorodisskij i  
Partnery", pat.pov. Ju.D.Kuznetsov, reg.№ 595

(72) Inventor: DZANG Euee-seon (KR),  
KIM Do-kioon (KR), VOO Sang-oak (KR)

(73) Proprietor:  
SAMSUNG EhLEKTRONIKS KO., LTD (KR)

(74) Representative:  
Kuznetsov Jurij Dmitrievich

(54) METHODS AND DEVICES FOR COMPRESSION AND RECOVERY OF ANIMATION PATH USING LINEAR APPROXIMATIONS

(57) Abstract:

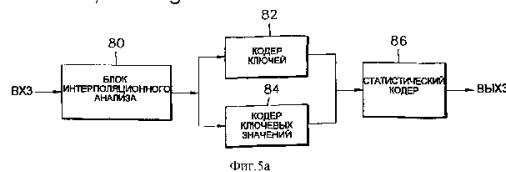
FIELD: animation of three-dimensional pictorial models.

SUBSTANCE: compression device has interpolation analysis unit for discriminating definite quantity of breakpoints on animation path and for outputting keys and key values corresponding to breakpoints; key coder; key value coder; and statistical coder. Analysis of breakpoints yields encoded bit stream having

minimal quantity of breakpoints.

EFFECT: enhanced efficiency of data compression in interpolator form for fast data transfer and storage.

20 cl, 35 dwg



R  
U  
2  
2  
3  
6  
7  
5  
1  
C  
2

C 2  
? 2 3 6 7 5 1

R U ? 2 3 6 7 5 1 C 2

R U 2 3 6 7 5 1 C 2

Область техники, к которой относится изобретение

Настоящее изобретение относится к анимации трехмерных (3D) графических моделей, а более конкретно к устройствам для сжатия и восстановления траектории анимации, которая применяется в анимации, с использованием линейной аппроксимации, а также к способам сжатия и восстановления траектории анимации, используемым в этих устройствах, и к форматам данных для этих устройств и способов.

Предшествующий уровень техники

В компьютерной 3D-анимации используют интерполяторы для выражения движения и поворота в пространстве, плавного преобразования (морфинга) модели, изменений цветов и т.д. моделируемого трехмерного объекта (3D-объекта).

На фиг.1 показана схема, поясняющая траекторию анимации в обычной 3D-анимации, причем вертикальная ось представляет ключевые значения (КЛЮЧЕВОЕ\_ЗНАЧЕНИЕ), а горизонтальная ось представляет ключи (КЛЮЧ).

Как показано на фиг.1, траектория 20 анимации изображает след анимации 3D-модели 10. Траектория 20 информации анимации является двухмерной кривой, изменяющейся во времени, как показано на фиг.1. Траекторию анимации можно представить множеством способов, пояснение которых приведено в главе 9 книги А.К.Джайна "Основы цифровой обработки изображений", опубликованной издательством "Прентис Холл" в 1989 г. (A.K.Jain, "Fundamentals of Digital Image Processing", Prentice Hall, 1989).

В выражении, получаемом с использованием интерполятора, траекторию 20 анимации, имеющую форму кривой, как показано на фиг.1, можно представить определенными прямыми линиями с использованием множества отрезков. Существенная информация в этом выражении включает в себя точки разрыва или вершины каждого определенного отрезка прямой линии. В данном случае точки разрыва или вершины выражены в виде точек на траектории 20 анимации, показанной на фиг.1. Используя линейную интерполяцию, можно восстановить исходную кривую по точкам разрыва.

На фиг.2 представлен пример выражения (скалярного интерполятора) траектории анимации, используемого в языке моделирования виртуальной реальности (ЯМВР) или в четвертой версии алгоритма сжатия подвижного изображения, разработанного Экспертной группой по кинематографии, т.е. алгоритма ЭГК-4 (MPEG-4). Обрабатываемая информация включает в себя ключи и ключевые значения, а линейная интерполяция осуществляется с использованием заданной информации.

Интерполяторы можно разделить, грубо говоря, на 6 видов: скалярные интерполяторы, позиционные интерполяторы, координатные интерполяторы, ориентационные интерполяторы, нормальные интерполяторы и цветовые интерполяторы. Среди них скалярные интерполяторы можно выразить так, как показано на фиг.2. Характеристики и функции 6-ти видов интерполяторов показаны в нижеследующей

таблице 1, причем все интерполяторы являются наборами заданных ключей и ключевых значений, соответствующих этим ключам.

			Таблица 1
5			
10			
15			
20			
25			
30			
35			
40			
45			
50			
55			
60			

На фиг.3 представлена условная схема для пояснения формата данных 3D-анимации и показан кодер 30, декодер 40, формат 50 файлов данных 3D-анимации. В данном случае формат 50 файлов 3D-анимации, выводимый из кодера 30, сформирован так, что в нем присутствуют данные модели, данные анимации, атрибуты, видео/текстура и звук.

На фиг.3 отмечено, что интерполятор соответствует данным анимации, которые эффективно выражают траекторию 3D-анимации. Данные 3D-анимации, выраженные с помощью ЯМВР или алгоритма MPEG-4, сформированы с информацией, показанной на фиг.3. Хотя для аудио-, видео- и 3D-моделей разработаны стандартизованные технологии сжатия, только технологии сжатия общего назначения, ориентированные на выражение, разработаны для интерполятора с целью определения траектории анимации. В анимации, включающей в себя аудио- и/или видео данные, объем данных для траекторий анимации наряду с 3D-моделями составляет большинство необходимого объема данных.

Следовательно, наряду с технологией сжатия 3D-моделей важна и технология сжатия траекторий анимации. Хотя двоичный формат для сцены (ДФС), соответствующий алгоритму MPEG-4, обеспечивает основной метод квантования и/или сжатия для анимации, этот метод отражает не технологию, предназначенную для интерполяторов, а технологию общего назначения, и имеет плохую рабочую характеристику сжатия. Это описано в статье Эюи С.Джэнга "Кодирование 3D-анимации: его история и основные положения" в сб. трудов Международной конференции и экспозиции по мультимедиа, проведенной в г. Нью-Йорк в 2000 г. (Eunee S.Jang "3D Animation Coding: its History and Framework", proceedings of the International

RU 2236751 C2 ? 2 3 6 7 5 1 C 2

Conference on multimedia and Expo held in New York city in 2000).

На фиг.4а и 4б представлены блок-схемы известных устройств для сжатия и восстановления траекторий анимации соответственно. Известное устройство для сжатия, показанное на фиг.4а, выполнено с блоком 60 скалярного квантования, а известное устройство для восстановления, показанное на фиг.4б, выполнено с блоком 70 скалярного деквантования. Исходная траектория анимации вводится в форме (ключ, ключевое значение) через входной вывод ВХ1 в блок 60 скалярного квантования, показанный на фиг.4а, и подвергается скалярному квантованию. Закодированный поток битов, который является результатом скалярного квантования, выводится через выходной вывод ВЫХ1. Блок 70 скалярного деквантования, показанный на фиг.4б, принимает этот закодированный поток битов через входной вывод ВХ2 и выводит данные в форме восстановленной траектории анимации (ключ, ключевое\_значение) через выходной вывод ВЫХ2.

Сжатие с помощью интерполятора в известном ДФС, соответствующем алгоритму MPEG-4, требует скалярного квантования, как показано на фиг.4а. Известный процесс сжатия, показанный на фиг.4а, применим не только к интерполяторам, но и ко всем элементам, которые требуют сжатия в ДФС. В порядке, обратном порядку сжатия, траектория анимации восстанавливается посредством блока 70 скалярного деквантования, и это происходит с использованием закодированного потока битов, вводимого в известное устройство для восстановления, показанное на фиг.4б. В устройствах, показанных на фиг.4а и 4б, ключи и ключевые значения интерполяторов сжимаются одинаковым образом, без учета характеристик каждого вида (интерполяторов), так что сжатие нельзя максимизировать.

Краткое изложение сущности изобретения

Чтобы решить вышеупомянутые проблемы, первая задача настоящего изобретения состоит в том, чтобы разработать устройство и способ сжатия траектории анимации с использованием линейной аппроксимации, в которых осуществляют эффективное сжатие данных анимации в форме интерполятора таким образом, что передача и запоминание данных осуществляются быстро.

Вторая задача настоящего изобретения состоит в том, чтобы разработать устройство и способ восстановления траектории анимации, предназначенные для эффективного восстановления данных сжатой траектории анимации.

Третья задача настоящего изобретения состоит в том, чтобы разработать формат данных для сжатия данных траектории анимации.

Для решения первой задачи настоящего изобретения предложено устройство для сжатия траектории анимации, имеющее блок интерполяционного анализа для выделения предварительно определенного количества точек разрыва из траектории анимации и вывода ключей и ключевых значений, соответствующих точкам разрыва, кодер ключей для кодирования ключей, выводимых

из блока интерполяционного анализа, кодер ключевых значений для кодирования ключевых значений, выводимых из блока интерполяционного анализа, и статистический кодер для статистического кодирования ключей и ключевых значений, которые закодированы в кодере ключей и кодере ключевых значений соответственно, и вывода закодированных потоков битов.

Для решения второй задачи настоящего изобретения предложено устройство для восстановления траектории анимации, имеющее статистический декодер для приема закодированного потока битов и статистического декодирования этого потока битов, декодер ключей для приема результата статистического декодирования и для декодирования ключей, декодер ключевых значений для приема результата статистического декодирования и для декодирования ключевых значений и блок интерполяционного восстановления для получения пустых ключевых значений путем линейной интерполяции на основании ключей и ключевых значений, декодированных в декодере ключей и декодере ключевых значений соответственно, и восстановления исходной траектории анимации.

Для решения первой задачи настоящего изобретения также предложен способ сжатия траектории анимации, включающий этапы, на которых выделяют предварительно определенное количество точек разрыва из исходной траектории анимации, выделяют ключи и ключевые значения с использованием выделенных точек разрыва и кодируют эти ключи и ключевые значения, и осуществляют статистическое кодирование закодированных ключей и ключевых значений для получения закодированных потоков битов.

Для решения второй задачи настоящего изобретения также предложен способ выделения точек разрыва траектории анимации, включающий этапы, на которых (а) выбирают две точки разрыва в обеих концевых точках исходной траектории анимации среди точек разрыва на траектории анимации, (б) выбирают одну точку разрыва среди остающихся точек разрыва, за исключением двух выбранных точек разрыва, (в) интерполируют ключевые значения остающихся точек разрыва, за исключением выбранных точек разрыва, с использованием выбранных точек разрыва, (г) формируют аппроксимированную траекторию анимации на основании выбранных точек разрыва и интерполированных ключевых значений, выбирают аппроксимированную траекторию анимации, которая имеет наименьшую разность траекторий между исходной траекторией анимации и этой аппроксимированной траекторией анимации, и выбирают точки разрыва, соответствующие выбранной траектории анимации, и (д) выбирают одни точки разрыва среди остающихся точек разрыва, за исключением точек разрыва, выбранных на этапах (а) и (б), и повторяют этапы (в)-(д) до тех пор, пока разность траекторий не станет меньше, чем допустимая разность.

Разность траекторий предпочтительно выражают суммой площадей трапецидлов или "скрученных" трапецидов, которые образованы исходной траекторией анимации

и аппроксимированной траекторией анимации.

Разность траекторий в ориентационном интерполяторе предпочтительно определяют как дифференциальный угол поворота в дифференциальном преобразовании поворота, представляющем собой разность между преобразованием поворота исходной траектории анимации и преобразованием поворота аппроксимированной траектории.

Для решения второй задачи настоящего изобретения также предложен способ восстановления траектории анимации, включающий в себя этапы, на которых осуществляют прием и статистическое декодирование закодированного потока битов, декодируют ключи и ключевые значения исходя из результата статистического декодирования и восстанавливают исходную траекторию анимации, получая пустые ключевые значения путем линейной интерполяции на основании декодированных ключей и ключевых значений.

Для решения третьей задачи настоящего изобретения предложен формат данных потока битов, который получен путем кодирования траектории анимации, причем этот формат данных имеет метку ключа для указания ключевых значений, оси которых выбраны среди ключевых значений, соответствующих координате x, у или z каждой точки разрыва траектории анимации, ключ типа "массив" для указания того, что, по меньшей мере, одно или более ключевых значений выбраны среди ключевых значений, соответствующих координате x, у или z каждой точки разрыва, и ключевые значения типа "массив" для указания ключевых значений, выбранных для каждой точки разрыва.

#### Краткое описание чертежей

Вышеупомянутые задачи и преимущества настоящего изобретения станут более очевидными из подробного описания предпочтительных конкретных вариантов осуществления изобретения, приводимого со ссылками на прилагаемые чертежи, где

на фиг.1 представлена схема, поясняющая траекторию анимации в обычной трехмерной анимации (3D-анимации),

на фиг.2 представлен пример выражения траектории анимации, используемого в языке моделирования виртуальной реальности (ЯМВР) или алгоритме MPEG-4,

на фиг.3 представлена условная схема, поясняющая формат данных 3D-анимации,

на фиг.4а и 4б представлены блок-схемы известных устройств для сжатия и восстановления траектории анимации соответственно,

на фиг.5а и 5б представлены блок-схемы устройств для сжатия и восстановления траектории анимации согласно настоящему изобретению соответственно,

на фиг.6 представлена блок-схема предпочтительного конкретного варианта осуществления устройства для сжатия согласно настоящему изобретению, показанного на фиг.5а,

на фиг.7 представлена блок-схема предпочтительного конкретного варианта осуществления устройства для восстановления согласно настоящему изобретению, показанного на фиг.5б,

- на фиг.8а - 8h представлены схемы, поясняющие предпочтительный конкретный вариант выделения точек разрыва с использованием линейной аппроксимации согласно настоящему изобретению,
- на фиг.9 представлена схема, поясняющая способ получения разности между действительной траекторией анимации и аппроксимированной траекторией анимации,
- на фиг.10 представлена схема процесса квантования, а более конкретно широко применяемого метода кодирования посредством дифференциальной импульсно-кодовой модуляции (ДИКМ),
- на фиг.11а - 11е представлены схемы, изображающие форматы закодированных потоков битов согласно настоящему изобретению,
- на фиг.12 представлена таблица, поясняющая процесс декодирования, выраженный в синтаксическом выражении согласно настоящему изобретению,
- на фиг.13а - 13д представлены графики экспериментальных последовательностей анимации для сравнения способа скатия согласно настоящему изобретению с известным способом скатия,
- на фиг.14а - 14f представлены схемы, поясняющие еще один предпочтительный конкретный вариант выделения точек разрыва с использованием сферической линейной аппроксимации согласно настоящему изобретению.
- Описание предпочтительных конкретных вариантов осуществления
- В нижеследующем описании настоящего изобретения употребляемые выражения интерполяторов являются выражениями, используемыми в ЯМВР и/или алгоритме MPEG-4, а области, в которых применяются интерполяторы, включают в себя компьютерные игры в диалоговом режиме, анимационные рекламные объявления и т.д.
- На фиг.5а представлена блок-схема устройства для сжатия траектории анимации согласно настоящему изобретению, включающего в себя блок 80 интерполяционного анализа, кодер 82 ключей (К-кодер 82), кодер 84 ключевых значений (КЗн-кодер 84) и статистический кодер 86. На фиг.5б представлена блок-схема устройства для восстановления траектории анимации согласно настоящему изобретению, включающего в себя статистический декодер 90, декодер 92 ключей, декодер 94 ключевых значений и блок 96 интерполяционного восстановления.
- Блок 80 интерполяционного анализа устройства для сжатия, показанного на фиг.5а, выбирает ключи и ключевые значения, подлежащие кодированию, и может быть сконструирован с учетом характеристик разных типов интерполяторов. Пример подробного анализа будет пояснен ниже.
- На фиг.6 представлена блок-схема предпочтительного варианта осуществления устройства для сжатия согласно настоящему изобретению, показанного на фиг.5а. Устройство для сжатия, показанное на фиг.6, включает в себя блок 80 интерполяционного анализа, имеющий блок 100 нормализации и блок 102 минимизации точек разрыва, кодер 82 ключей, имеющий квантователь 104, работающий в режиме дифференциальной

R U ? 2 3 6 7 5 1 C 2

импульсно-кодовой модуляции (ДИКМ-квантователь 104), кодер 84 ключевых значений, имеющий блок 106 квантования посредством ДИКМ (блок 106 ДИКМ-квантования), и статистический кодер 86.

На фиг.7 представлена блок-схема предпочтительного варианта осуществления устройства для восстановления согласно настоящему изобретению, показанного на фиг.5б. Это устройство для восстановления включает в себя статистический декодер 90, декодер 92 ключей, имеющий деквантователь 126, работающий в режиме ДИКМ (ДИКМ-деквантователь 126), декодер 94 ключевых значений, имеющий ДИКМ-деквантователь 128, и блок интерполяционного восстановления, имеющий блок 95 интерполяционного восстановления ключей и ключевых значений.

Интерполяционное выражение, подаваемое на вход блока 80 интерполяционного анализа, включает в себя ключи (K) и ключевые значения (K3н). В устройстве для сжатия, показанном на фиг.6, блок 80 интерполяционного анализа регулирует количество точек разрыва таким образом, что траекторию анимации, которую вводят через входной вывод BX3, можно выразить с помощью минимального количества точек разрыва. Если исходное количество точек разрыва траектории анимации составляет N, то количество точек разрыва, выводимых из блока 80 интерполяционного анализа, регулируется до величины M ( $M \leq N$ ). То есть, блок 80 интерполяционного анализа выделяет точки разрыва из исходной траектории анимации, выводимой на входной вывод BX3.

В интерполаторе ключ и ключевое значение можно соответственно нормализовать и использовать. Для этого блок 100 нормализации нормализует каждый из ключей и каждое из ключевых значений на исходной траектории анимации, выводимые через входной вывод BX3, и выводит нормализованные результаты в блок 102 минимизации точек разрыва. Ключ, поддерживаемый ЯМВР, имеет значение в диапазоне между 0 и 1 включительно.

Способ, которым блок 80 интерполяционного анализа регулирует количество точек разрыва, может быть определен таким образом, что минимизируется разность траектории анимации, образованной подобранными точками разрыва, и исходной траектории анимации. Для этого блок 102 минимизации точек разрыва выдает точки разрыва, выделенные исходя из нормализованного ключа и нормализованных ключевых значений, которые выводятся из блока 102 нормализации, таким образом, что минимизируется количество выделенных точек разрыва. Например, в случае позиционного интерполятора, блок 102 минимизации точек разрыва определяет точки разрыва таким образом, что минимизируется площадь, представляющая собой погрешность между действительной траекторией и квантованной траекторией, которая определяется блоком 80 интерполяционного анализа.

Теперь будет пояснено выделение точек разрыва с использованием линейной

аппроксимации, осуществляющейся в блоке 102 минимизации точек разрыва, показанном на фиг.6.

На фиг.8а - 8h представлены схемы, поясняющие предпочтительный вариант выделения точек разрыва с использованием линейной аппроксимации согласно настоящему изобретению. Каждая точка на траекториях анимации представляет собой точку разрыва. На фиг.8а представлена исходная траектория анимации. На фиг.8б показан процесс поиска обеих концевых точек (A, B) этой траектории. На фиг.8с показан процесс выбора точки разрыва, которая является ближайшей к исходной траектории. На фиг.8д показаны точки разрыва A, B и C, которые выбраны первыми. На фиг.8е показан процесс выделения вторых точек разрыва, которые близки к исходной траектории. На фиг.8ф показаны точки разрыва A, B, C, D, E и F, которые выбраны вторыми. На фиг.8г показаны точки разрыва A, B, C, D, E и F, которые выбраны третьими. На фиг.8h показаны точки разрыва A, B, C, D, E и F, которые выбраны четвертыми.

Когда исходная траектория анимации задана такая, как показано на фиг.8а, повторяется процесс поиска точек разрыва ( $M$  точек разрыва,  $2 \leq M \leq N$ ), представляющих кривую анимации, которая имеет наименьшее отличие (наименьшую разность) от исходной траектории, среди точек разрыва ( $N$ ) на исходной траектории анимации, при этом обе концевые точки A и B принимаются за начальные точки. Этот процесс показан на фиг.8б - 8h. В то же время можно повторять выделение точек разрыва до тех пор, пока не будет определено, что кривая аппроксимации достаточно близка к исходной кривой.

Со ссылками на прилагаемые чертежи способ, при котором используется разность площадей, описан ниже в качестве способа получения разности между кривой аппроксимации и действительной кривой.

На фиг.9 представлена схема, поясняющая способ получения разности между действительной траекторией анимации и аппроксимированной траекторией анимации. Обращаясь к фиг.9, отмечаем, что разность между этими двумя траекториями выражается трапецидом или "скрученным" трапецидом. Уравнение (1) для получения площади трапецида и уравнение (2) для получения площади "скрученного" трапецида приведены ниже. Разность между действительной траекторией 200 анимации и аппроксимированной траекторией 210 анимации можно выразить суммой площади трапецидов и площади "скрученных" трапецидов. Сумма этих площадей является разностью ( $D_{пл}$ ) площадей между этими двумя траекториями, как показано в уравнении (3).

$$D_{\text{трапецида}} = (a + b)h / 2, \quad (1)$$

$$D_{\text{скрученного трапецида}} = (1/2) \cdot ((a^2 + b^2) / (a+b)) \cdot h, \quad (2)$$

$$D_{пл} = \sum_i D_{\text{трапецида}} + \sum_j D_{\text{скрученного трапецида}} \quad (3)$$

Точки разрыва выделяют таким образом, что минимизируется разность ( $D_{пл}$ ) площадей, получаемая с помощью уравнения (3). Таким образом, блок 102 минимизации точек разрыва выделяет точки разрыва. Существенные точки разрыва траектории

R U 2 2 3 6 7 5 1 C 2

анимации выделяются путем пропускания через блок 80 интерполяционного анализа. Когда необходима обработка без потерь, количество (M) выделенных точек разрыва может быть таким же, как количество исходных точек разрыва.

В позиционном интерполяторе ключевые значения представляют положение в 3D-пространстве, имеющем оси X, Y и Z, а траектория анимации представляется тремя кривыми в осях X, Y и Z. Блок 80 интерполяционного анализа может выделять точки разрыва по каждой оси, и в то же время точки разрыва на каждой оси могут отличаться от точек разрыва на других осях. В нижеследующей таблице 2 приведен результат выделения новых точек разрыва в блоке 80 интерполяционного анализа из действительной траектории, имеющей 8 точек разрыва (T0, T1, T2, T3, T4, T5, T6 и T7).

Таблица 2								
Точки разрыва	T0	T1	T2	T3	T4	T5	T6	T7
кзн <sub>x</sub>	○	X	X	X	○	X	X	C
кзн <sub>y</sub>	○	X	○	X	C	X	X	C
кзн <sub>z</sub>	○	X	○	X	X	○	X	C
Ключ	○	X	○	X	○	○	X	C
Метка_ключа	7	-	6	-	3	4	-	7

В таблице 2 "метка\_ключа" обозначает метку ключа, а кзн<sub>x</sub>, кзн<sub>y</sub> и кзн<sub>z</sub> обозначают ключевые значения по осям X, Y и Z соответственно. "О" указывает, что выбрано ключевое значение, соответствующее координате x, у или z каждой точки разрыва. Например, ключевое значение кзн<sub>x</sub> выбрано для точки T0 разрыва. "X" указывает, что ключевое значение, соответствующее какой-либо оси в каждой точке разрыва, не выбрано. Метка ключа указывает ключевые значения, оси которых выбраны, среди ключевых значений для осей X, Y, Z в каждой точке разрыва. Например, "7" указывает, что все ключевые значения для всех осей выбраны, а "6" указывает, что выбраны все ключевые значения, за исключением ключевого значения кзн<sub>x</sub> для оси X.

Как показано в таблице 2, точки T1, T3 и T6 разрыва относятся к необязательным точкам разрыва (помечены прочерком, "-", в графе "метка\_ключа"), потому что в точках разрыва не выбрано ни одно ключевое значение для осей X, Y и Z. T0 и T7 являются точками разрыва на обоих концах траектории, в которых выбраны ключевые значения для всех осей. В T2 не выбрано ключевое значение для оси X, но выбраны ключевые значения по другим осям. Следовательно, блок 80 интерполяционного анализа определяет ключи, соответствующие 5-ти точкам разрыва, вместо исходных 8-ми точек разрыва, а также определяет ключевые значения для каждой оси, соответствующие выбранным точкам разрыва. В таблице 2 показано, что необходимы 5 ключей и 11 ключевых значений. Соответствующее соотношение между ключом и ключевыми значениями можно представить меткой ключа (метка\_ключа), которая передается дополнительно. Таким образом, количество ключей и ключевых значений уменьшается с 32 у действительной траектории (4.8=32) до 16 ключей и ключевых значений плюс одна дополнительная метка ключа. Следовательно, посредством такого процесса достигается упрощение в выражении.

На фиг.14а - 14f представлены схемы,

поясняющие еще один предпочтительный вариант выделения точек разрыва с использованием сферической линейной аппроксимации согласно настоящему изобретению. На этих чертежах показан способ уменьшения количества точек разрыва (также называемых ключевыми кадрами) в ориентационном интерполяторе.

На фиг.14а показаны ключевые значения (=Q<sub>0</sub>, Q<sub>1</sub>, Q<sub>2</sub>, ..., Q<sub>n</sub>) в каждой точке разрыва применительно к количеству n+1 временных точек на исходной траектории анимации, и эти ключевые значения отмечены черными точками. Как показано на фиг.14б, первыми выбирают две точки разрыва (=Q<sub>0</sub>, Q<sub>n</sub>), соответствующие двум концам траектории анимации. Эти выбранные точки разрыва показаны как белые точки.

На фиг.14с показан процесс выбора точки разрыва (третьей точки разрыва) среди остающихся точек разрыва, за исключением двух выбранных концевых точек разрыва. В этот момент количество способов выбора одной точки разрыва составляет (n-1). На фиг.14с показан пример, в котором выбираются два кандидата (=Q<sub>1</sub>, Q<sub>k</sub>). Для каждого из (n-1) кандидатов с использованием трех уже выбранных точек разрыва осуществляют сферическую линейную интерполяцию по ключевым значениям точек разрыва, которые не выбраны. Сравнивая исходную траекторию анимации и каждую из (n-1) траекторий-кандидатов анимации, полученных путем интерполяции, выбирают траекторию-кандидата анимации, имеющую наименьшую разность траекторий. Точку разрыва, соответствующую выбранной траектории-кандидату анимации, выбирают среди (n-1) точек-кандидатов разрыва. В качестве примера на фиг.14d изображен вариант, в котором выбрана траектория кандидата 2, как показано на фиг.1. Погрешность между траекториями получают путем использования понятия средней погрешности E<sub>ср</sub>.

Обращаясь к фиг.14е, отмечаем, что четвертую точку разрыва выбирают, осуществляя процесс, поясненный со ссылками на фиг.14с и 14д, после выбора точки-кандидата разрыва среди остающихся точек разрыва, за исключением трех выбранных точек разрыва, показанных на фиг.14д. В качестве примера, на фиг.14f показано, что выбран кандидат 1. Повторяя процесс выбора точек-кандидатов, показанный на фиг.14е, до тех пор, пока средняя погрешность не станет меньше, чем допустимая погрешность, выбирают точки разрыва, количество которых является таким же, как количество точек разрыва исходной траектории, или меньшим, чем количество точек разрыва исходной траектории.

Ниже приведено пояснение средней погрешности E<sub>ср</sub>, которая упоминалась выше. Погрешность квантования определяется как дифференциальный угол поворота в дифференциальном преобразовании поворота, являющемся разностью между преобразованием поворота исходной траектории анимации и преобразованием поворота восстановленной траектории анимации. То есть, в предположении, что ( $\bar{E}$ ,  $\theta$ ) обозначает ключевое значение

узла ориентационного интерполятора, а  $(\bar{r}', \theta')$  обозначает ключевое значение, восстановленное в блоке 96 интерполяционного восстановления (вектор  $\bar{r}$  обозначает ось поворота,  $\theta$  обозначает величину поворота, и что эта величина поворота удовлетворяет условию  $\theta \in [-\pi, \pi]$ ), когда осуществляется преобразование поворота из произвольного положения  $\bar{x}$  в  $\bar{y}$  и  $\bar{y}'$  в 3D-пространстве с использованием  $(\bar{r}, \theta)$  и  $(\bar{r}', \theta')$ , возникающую погрешность квантования вычисляют как разность между  $\bar{y}$  и  $\bar{y}'$ . Таким образом, если  $e(\bar{x})$  представляет вектор погрешности квантования, то  $e(\bar{x}) = \bar{y} - \bar{y}'$ . В кватернионном выражении  $X$ ,  $Y$  и  $Y'$  определяются, как следующие уравнения (4):

$$\begin{cases} X = (0, \bar{x}) \\ Y = (0, \bar{y}) \\ Y' = (0, \bar{y}') \end{cases} \quad (4)$$

Если  $Q$  и  $Q'$  обозначают кватернионные выражения для  $(\bar{r}, \theta)$  и  $(\bar{r}', \theta')$  соответственно, которые представляют преобразование поворота, то отсюда вытекают следующие уравнения (5):

$$\begin{aligned} Y &= Q * X * Q^* \\ X &= Q^* * Y * Q \end{aligned} \quad (5)$$

Здесь  $A * B$  обозначает умножение кватернионов  $A$  и  $B$ , а  $A^*$  обозначает кватернион, сопряженный с  $A$ . Отсюда вытекает следующее уравнение (6):

$$Y' = Q'^* * X * Q'' = Q'^* * Q'' * Y * Q * Q'' = Q'' * Y * Q'' \quad (6)$$

Здесь  $Q''$  - это величина, необходимая для указания связи преобразования поворота между  $\bar{y}$  и  $\bar{y}'$  и определяемая следующим уравнением (7):

$$Q'' = Q'^* * Q^* \quad (7)$$

Следовательно, если  $\theta''$  обозначает дифференциальный угол поворота между  $\bar{y}$  и  $\bar{y}'$ , то  $\theta''$  можно получить, пользуясь уравнением преобразования кватерниона и уравнения (7), как показано в следующем уравнении (8):

$$\theta'' = 2 \cos^{-1} q_0'' = 2 \cos^{-1}(Q' * Q), \theta'' \in [0, \pi], q_0'' = Q'' * Q, \quad (8)$$

(где  $*$  обозначает операцию скалярного произведения).

Уравнение (8) указывает мгновенную погрешность квантования, возникающую в предварительно определенный момент, по всем точкам разрыва траектории анимации. Чтобы вывести уравнение для получения погрешности квантования всех интервалов анимации, мгновенную погрешность квантования в предварительно определенный момент  $t$  можно выразить в виде следующего уравнения (9):

$$e(t) = 2 \arccos(Q(t) * Q'(t)) \quad (9)$$

Если уравнение (9) применяется посредством ориентационных

интерполяторов ко всем интервалам анимации с точками разрыва, то можно вывести среднюю погрешность  $E_{cp}$  и максимальную погрешность  $E_{max}$  для всего интервала  $[t_0, t_L]$  в виде следующих уравнений (10):

$$\begin{cases} E_{\Phi} = \sqrt{\frac{1}{t_L - t_0} \int_0^L e^2(t) dt} \\ E_{max} = \max |e(t)| \end{cases} \quad (10)$$

Здесь нужно отметить, что для получения

$E_{cp}$  сначала получают частичную сумму  $E_{\Phi}^i$  для интервала  $[t_{i-1}, t_i]$ , пользуясь следующим уравнением (11):

$$E_{\Phi}^i = \int_{t_{i-1}}^{t_i} e^2(t) dt = 4 \int_{t_{i-1}}^{t_i} \arccos^2(Q(t) * Q'(t)) dt \quad (11)$$

Вместе с тем, выводят также следующее уравнение (12):

$$4 \arccos^2(Q(t) * Q'(t)) = \phi^2(\alpha), t = t_{i-1} + \alpha(t_i - t_{i-1}) \quad (12)$$

А отсюда выводят следующее уравнение (13):

$$E_{\Phi}^i = (t_i - t_{i-1}) \int_0^1 \phi^2(\alpha) d\alpha \quad (13)$$

Поскольку трудно получить определенный интеграл функции  $\phi^2(\alpha)$  в интервале между 0 и 1, осуществляют аппроксимацию, показанную в следующих уравнениях (14):

$$\begin{aligned} \phi(\alpha) &\cong \phi(0) + \alpha(\phi(1) - \phi(0)) \\ \phi^2(\alpha) &\cong \phi^2(0) + \alpha^2(\phi(1) - \phi(0))^2 + 2\alpha\phi(0)(\phi(1) - \phi(0)) \end{aligned} \quad (14)$$

Здесь

$$\cos \frac{\phi(0)}{2} = Q(t_{i-1}) * Q'(t_{i-1}), \cos \frac{\phi(1)}{2} = Q(t_i) * Q'(t_i) \quad (15)$$

Используя аппроксимированную функцию (14), получают частичную сумму  $E_{cp}^i$  в виде следующего уравнения (16):

$$E_{\Phi}^i \cong \frac{1}{3} (t_i - t_{i-1}) (\phi^2(0) + \phi^2(1) + \phi(0)\phi(1)) \quad (16)$$

Эти уравнения можно перегруппировать в виде следующего уравнения (17):

$$E_{\Phi}^i \cong \frac{4}{3} (t_i - t_{i-1}) (\arccos^2(Q(t_{i-1}) * Q'(t_{i-1})) + \arccos^2(Q(t_i) * Q'(t_i))) + \arccos(Q(t_{i-1}) * Q'(t_{i-1})) \arccos(Q(t_i) * Q'(t_i)) \quad (17)$$

Чтобы получить среднюю погрешность  $E_{cp}$ , суммируют частичную сумму  $E_{cp}^i$  по всему интервалу  $[t_0, t_L]$ , как показано в следующем уравнении (18):

$$E_{\Phi} \cong \sqrt{\frac{1}{t_L - t_0} \sum_{i=0}^L E_{\Phi}^i} \quad (18)$$

Чтобы получить максимальную погрешность  $E_{max}$ , выбирают максимальное значение среди значений максимальной погрешности  $E_{max}^i$  в каждом интервале  $[t_{i-1}, t_i]$ , которое получают с помощью следующего уравнения (19):

$$E_{max}^i \cong \max |e(t)| = \max 2 |\arccos(Q(t) * Q'(t))| \quad (19)$$

Пользуясь вышеописанной функцией аппроксимации, можно аппроксимировать  $E_{max}$ , как показано в следующем уравнении (20):

R U ? 2 3 6 7 5 1 C 2

$$E_{\max}^i \equiv \max(\phi(0), \phi(1)) = \max(2|\arccos(Q(t_{i-1}) \cdot Q'(t_{i-1}))|, 2|\arccos(Q(t_i) \cdot Q'(t_i))|) \quad (20)$$

Максимальную погрешность  $E_{\max}$  во всем интервале  $[t_0, t_L]$  выражают следующим уравнением (21):

$$E_{\max} = \max E_{\max}^i, \text{ для } i = 1, 2, \dots, L \quad (21)$$

Блок 80 интерполяционного анализа, показанный на фиг.6, посыпает информацию 114 о количестве ключей, минимальное значение и максимальное значение нормализованных ключевых значений, разрешения ключа и ключевого значения, а также метку ключа в статистический кодер 86. В то же время блок 80 интерполяционного анализа посыпает ключи 110 в кодер 82 ключей и посыпает информацию о ключевых значениях, включающую в себя минимальное значение и максимальное значение нормализованных ключевых значений, в кодер 84 ключевых значений.

Ниже приведено пояснение кодирования ключей и ключевых значений в кодере 82 ключей и кодере 84 ключевых значений. Ключи и ключевые значения сжимаются методом ДИКМ-квантования в кодере 82 ключей и кодере 84 ключевых значений соответственно. Квантованная информация вместе с другой информацией посыпается в статистический кодер 86 и выводится в виде потока битов, подвергнутого окончательному кодированию сжатием, через выходной вывод ВЫХ3. Например, ДИКМ-квантователь 104 кодера 82 ключей осуществляет кодирование методом ДИКМ-квантования (т.е. кодирует разность между текущим значением и непосредственным предыдущим значением) ключа, посыпаемого из блока 80 интерполяционного анализа, и выводит закодированный результат в статистический кодер 86. В то же время ДИКМ-квантователь 106 кодера 84 ключевых значений осуществляет кодирование методом ДИКМ-квантования (т.е. кодирует разность между текущим значением и непосредственным предыдущим значением) ключевых значений, а также минимального значения и максимального значения ключевых значений, посыпаемых из блока 80 интерполяционного анализа, и выводит закодированный результат в статистический кодер 86.

На фиг.10 представлена схема процесса квантования и проиллюстрирован широко применяемый метод ДИКМ. Обращаясь к фиг.10, отмечаем, что в случае ключа (k) окончательная погрешность (e) подвергается сжатию квантованием, как показывает следующее уравнение (22). Фиг.10 поясняет уравнение (22). То есть, получают разность ( $d_1$ ) между непосредственным предыдущим значением ( $K_{i-1}$ ) и его предыдущим значением ( $K_{i-2}$ ). Значение разности ( $d_1$ ) прибавляют к непосредственному предыдущему значению ( $K_{i-1}$ ) для формирования аппроксимированного значения ( $K_i$ ). Затем получают окончательную погрешность (e) как разность между аппроксимированным значением ( $K_i$ ) и текущим значением ( $K_i$ ). Кстати, погрешность (e) можно легко получить как разность между текущим значением ( $K_i$ ) и непосредственным предыдущим значением ( $K_{i-1}$ ).

$$\begin{aligned} K_i &= 0, \text{ для } i < 0 \\ d_1 &= K_{i-1} - K_{i-2}, \\ \hat{K}_i &= K_{i-1} + d_1, \\ d_2 &= K_i - \hat{K}_i, \\ e &= d_2 \end{aligned} \quad (22)$$

На фиг.11а - 11е представлены схемы, показывающие форматы закодированных потоков битов согласно настоящему изобретению. На фиг.11а показан формат закодированного потока битов согласно настоящему изобретению, который включает в себя количество ключей (кол\_ключей) разрешение ключей (разр\_k), разрешение ключевых значений (разр\_кзн), массив ([мин/макс]) 230 минимального значения и максимального значения ключевых значений, массив ([метка\_ключа]) 232 меток ключей, массив ([ключ]) 234 ключей и массив ([кзн]) 236 ключевых значений. Здесь квадратные скобки [] обозначают массив. На фиг.11б показан формат массива 230 минимального значения и максимального значения ключевых значений, на фиг.11с показан формат массива 232 меток ключей, на фиг.11д показан формат массива 234 ключей, а на фиг. 11е показан формат массива 236 ключевых значений.

Массив 230 минимального значения и максимального значения ключевых значений, показанный на фиг.11а, сформирован из минимальных значений для осей X, Y и Z (мин\_x, мин\_y и мин\_z) и максимальных значений для осей X, Y и Z (макс\_x, макс\_y и макс\_z). Массив 232 меток ключей сформирован из n меток ключей (метка\_0\_ключа, метка\_1\_ключа, метка\_2\_ключа, метка\_3\_ключа, ..., метка\_n-1\_ключа). Здесь n обозначает количество ключей. Массив 234 ключей сформирован из p ключей (ключ\_0, ключ\_1, ключ\_2, ключ\_3, ..., ключ\_n-1). Здесь данные ключей и меток ключей для точек разрыва, которые классифицированы как необязательные, например - точки разрыва T1, T3 и T6 из таблицы 2, можно исключить из массива ключей или массива меток ключей. Таким образом, массив ключей и/или массив меток ключей может состоять из данных, количество которых меньше количества (n) исходных ключей. Массив 236 ключевых значений сформирован из r первых ключевых значений (кзн\_X\_0, кзн\_X\_1, ..., кзн\_X\_p-1), q вторых ключевых значений (кзн\_Y\_0, кзн\_Y\_1, ..., кзн\_Y\_q-1) и г третьих ключевых значений (кзн\_Z\_0, кзн\_Z\_1, ..., кзн\_Z\_r-1). Здесь p≤n, q≤n и g≤n.

Между тем, устройство и способ восстановления траектории анимации согласно настоящему изобретению реализуются как инверсия процесса сжатия. Статистический декодер 90, показанный на фиг.7, принимает закодированный поток битов через входной вывод BX4, осуществляет статистическое декодирование этого потока битов и выводит результат статистического декодирования как в декодер 92 ключей, так и в декодер 94 ключевых значений. Декодер 92 ключей и декодер 94 ключевых значений принимают квантованные ключи и ключевые значения соответственно,

R U ? 2 3 6 7 5 1 C 2

которые являются результатом статистического декодирования, и, используя дополнительную информацию, например метку ключа, восстанавливают данные, имевшиеся перед квантованием. То есть, декодер 92 ключей принимает результат статистического декодирования и декодирует ключи, а декодер 94 ключевых значений принимает результат статистического декодирования и декодирует ключевые значения. Для этого декодер 92 ключей может быть выполнен в виде ДИКМ-деквантователя 126, а декодер 94 ключевых значений может быть выполнен в виде ДИКМ-деквантователя 128. Каждый ДИКМ-деквантователь - 126 или 128 - декодирует вводимые данные, декодированные в статистическом декодере 90, методом ДИКМ-деквантования.

Блок 96 интерполяционного восстановления принимает данные, декодированные в декодере 92 ключей и декодере 94 ключевых значений, и восстанавливает исходную траекторию анимации. Блок 96 интерполяционного восстановления принимает информацию 124 о количестве ключей, минимальное значение и максимальное значение нормализованных ключевых значений и флаги ключей из статистического декодера 90,ключи 120 из декодера 92 ключей и информацию 122 о ключевых значениях из декодера 94 ключевых значений. Блок 96 интерполяционного восстановления восстанавливает пустые ключевые значения посредством линейной интерполяции, пользуясь вводимой информацией. Например, в таблице 2 ключевое значение для оси X в точке T2 разрыва можно восстановить посредством линейной интерполяции, пользуясь ключевыми значениями для точек T0 и T4. Для этого блок 96 интерполяционного восстановления может быть выполнен в виде блока 95 интерполяционного восстановления ключей и ключевых значений. Восстановленная траектория анимации в форме (ключ, ключевые значения) выводится через выходной вывод ВЫХ4.

На фиг.12 представлена таблица, поясняющая процесс декодирования, выраженный на языке программирования согласно настоящему изобретению. Этую таблицу также называют синтаксисом потоков битов.

Обращаясь к прилагаемым чертежам, отмечаем, что способ сжатия траектории анимации с использованием линейной аппроксимации согласно настоящему изобретению теперь будет сравниваться с известным способом интерполяционного сжатия с использованием ДФС в соответствии с алгоритмом МРЕ6-4.

На фиг.13а - 13d представлены графики экспериментальных последовательностей анимации для сравнения способа сжатия согласно настоящему изобретению с известным способом сжатия. Вертикальная ось представляет степени искажения, горизонтальная ось представляет количество битов, способ сжатия согласно настоящему изобретению отображен толстыми линиями, а известный способ сжатия отображен пунктирными линиями. На фиг.13а - 13d показаны результаты вариантов моделирования, в каждом из которых

применена отличающаяся траектория анимации. Способ согласно настоящему изобретению показывает значительно лучшие результаты в порядке фиг.13б - 13а - 13д - 13с. При одних и тех же скоростях передачи битов способ сжатия согласно настоящему изобретению показал намного лучшее качество изображения, чем известный способ с использованием ДФС, а при одном и том же качестве изображения способ сжатия согласно настоящему изобретению показал гораздо больший коэффициент сжатия.

Как описано выше, интерполяционное выражение траектории анимации используется для упрощения этой траектории с использованием точек разрыва. Однако в известном способе интервал между каждым ключом сформирован равномерно в выражении траектории, и поэтому происходит избыточная дискретизация (передискретизация) точек разрыва. Вместе с тем, в настоящем изобретении получается упрощено закодированный поток битов, имеющий минимальное количество точек разрыва. Сжатие методом ДИКМ (ДИКМ-сжатие) поддерживает более высокую линейную корреляцию между точками разрыва, и поэтому обеспечивается эффективное сжатие с использованием этой более высокой корреляции.

#### Формула изобретения:

1. Устройство для сжатия траектории анимации, содержащее блок анализа интерполятора для выделения предварительно определенного количества точек разрыва из траектории анимации и выдачи ключей и ключевых значений, соответствующих точкам разрыва, кодер ключей для кодирования ключей, выдаваемых из блока анализа интерполятора, кодер ключевых значений для кодирования ключевых значений, выдаваемых из блока анализа интерполятора, и статистический кодер для статистического кодирования ключей и ключевых значений, которые закодированы в кодере ключей и кодере ключевых значений соответственно, и вывода закодированных потоков битов.

2. Устройство по п.1, в котором блок анализа интерполятора определяет количество точек разрыва таким образом, что минимизируется разность между аппроксимированной траекторией анимации, которую получают с помощью предварительно определенного количества точек разрыва, выделенных среди точек разрыва исходной траектории анимации, и исходной траекторией анимации.

3. Устройство по п.1, в котором кодер ключей кодирует ключи,ываемые из блока анализа интерполятора, методом квантования посредством дифференциальной импульсно-кодовой модуляции (ДИКМ-квантования).

4. Устройство по п.1, в котором кодер ключевых значений кодирует ключевые значения, выдаваемые из блока анализа интерполятора, методом ДИКМ-квантования.

5. Устройство по п.1, в котором принимают обе концевые точки исходной траектории анимации за начальные точки, блок анализа интерполятора находит среди точек разрыва точки разрыва, представляющие аппроксимированную траекторию анимации, которая имеет наименьшую разницу от

R U 2 2 3 6 7 5 1 C 2

исходной траектории анимации, и этот процесс повторяется до тех пор, пока аппроксимированная траектория анимации не станет близкой к исходной траектории.

6. Устройство по п.5, в котором точки разрыва определяются таким образом, что разность площадей между исходной траекторией анимации и аппроксимированной траекторией анимации является наименьшей.

7. Устройство по любому из п.3 или 4, в котором ДИКМ-квантование представляет собой кодирование разности между текущим значением и непосредственным предыдущим значением или представляет собой кодирование разности между текущим значением и аппроксимированным значением, которое получено путем прибавления непосредственного предыдущего значения к разности между непосредственным предыдущим значением и его предыдущим значением.

8. Устройство для восстановления траектории анимации, содержащее статистический декодер для приема закодированного потока битов и статистического декодирования этого потока битов, декодер ключей для приема результата статистического декодирования и для декодирования ключей, декодер ключевых значений для приема результата статистического декодирования и для декодирования ключевых значений, и блок восстановления интерполятора для получения ключевых значений путем линейной интерполяции на основании ключей и ключевых значений, декодированных в декодере ключей и декодере ключевых значений соответственно, и восстановления исходной траектории анимации.

9. Устройство по п.8, в котором декодер ключей декодирует результат статистического декодирования, который вводится из статистического декодера, методом ДИКМ-деквантования.

10. Устройство по п.8, в котором декодер ключевых значений декодирует результат статистического декодирования, который вводится из статистического декодера, методом ДИКМ-деквантования.

11. Способ восстановления траектории анимации, включающий в себя этапы, на которых осуществляют прием и статистическое декодирование закодированного потока битов, декодируют ключи и ключевые значения, исходя из результата статистического декодирования, и восстанавливают исходную траекторию анимации, получая ключевые значения путем линейной интерполяции на основании декодированных ключей и ключевых значений.

12. Способ сжатия траектории анимации, включающий в себя этапы, на которых выделяют предварительно определенное количество точек разрыва из траектории анимации, выделяют ключи и ключевые значения с использованием выделенных точек разрыва и кодируют эти ключи и ключевые значения, и осуществляют статистическое кодирование закодированных ключей и ключевых значений для получения закодированных потоков битов, при этом количество точек разрыва регулируется таким образом, чтобы разность между траекторией анимации, сформированной выделенными

точками разрыва и исходной траекторией анимации, минимизируется.

13. Способ по п.12, в котором, дополнительно, данные располагают в закодированных потоках битов в виде формата, содержащего метки ключа в виде массива для указания ключевых значений, оси которых выбраны среди ключевых значений, соответствующих координате x, у или z каждой точки разрыва траектории анимации, ключи в виде массива для указания того, что, по меньшей мере, одно или более ключевых значений выбраны среди ключевых значений, соответствующих координате x, у или z каждой точки разрыва, и ключевые значения в виде массива, выбранных для каждой точки разрыва.

14. Способ по п.13, в котором формат дополнительного содержит минимальные значения ключевых значений, соответствующих осям X, Y и Z координат, и максимальные значения ключевых значений, соответствующих осям X, Y и Z координат.

15. Способ по п.13, в котором массив меток ключа включает в себя M данных, а M равно количеству исходных ключей или меньше этого количества.

16. Способ по п.13, в котором массив ключей включает в себя M данных, а M равно количеству исходных ключей или меньше этого количества.

17. Способ по п.13, в котором ключевые значения включают в себя p ( $\leq n$ ) первых ключевых значений, соответствующих оси x, q ( $\leq n$ ) вторых ключевых значений, соответствующих оси y, и r ( $\leq n$ ) третьих ключевых значений, соответствующих оси z, а n обозначает количество ключей.

18. Способ выделения точек разрыва траектории анимации, включающий этапы, на которых (а) выбирают две точки разрыва в обеих концевых точках исходной траектории анимации среди точек разрыва на траектории анимации, (б) выбирают одну точку разрыва среди остающихся точек разрыва, за исключением двух выбранных точек разрыва,

(в) интерполируют ключевые значения остающихся точек разрыва, за исключением выбранных точек разрыва, с использованием выбранных точек разрыва, (г) формируют аппроксимированную траекторию анимации на основании выбранных точек разрыва и интерполированных ключевых значений, выбирают аппроксимированную траекторию анимации, которая имеет наименьшую

разность траекторий между исходной траекторией анимации и этой аппроксимированной траекторией анимации, и выбирают точки разрыва, соответствующие выбранной траектории анимации, и (д) выбирают одни точки разрыва среди остающихся точек разрыва, за исключением точек разрыва, выбранных на этапах (а) и (б), и повторяют этапы (в)-(д) до тех пор, пока разность траекторий не станет меньше, чем допустимая разность.

19. Способ по п.18, при котором разность траекторий выражают суммой площадей трапецидальных или скрученных трапецидальных, которые образованы исходной траекторией анимации и аппроксимированной траекторией анимации.

20. Способ по п.18, при котором разность траекторий в ориентационном интерполяторе определяют как дифференциальный угол

R U ? 2 3 6 7 5 1 C 2

R U 2 2 3 6 7 5 1 C 2

поворота в дифференциальном преобразовании поворота, представляющем собой разность между преобразованием

поворота исходной траектории анимации и преобразованием поворота аппроксимированной траектории.

5

10

15

20

25

30

35

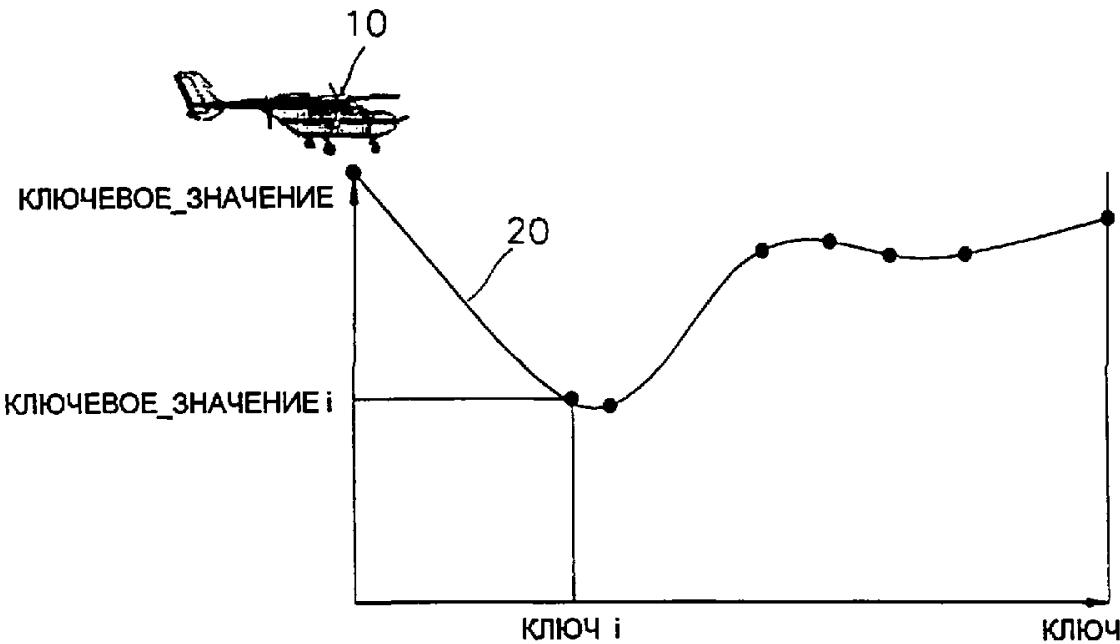
40

45

50

55

60



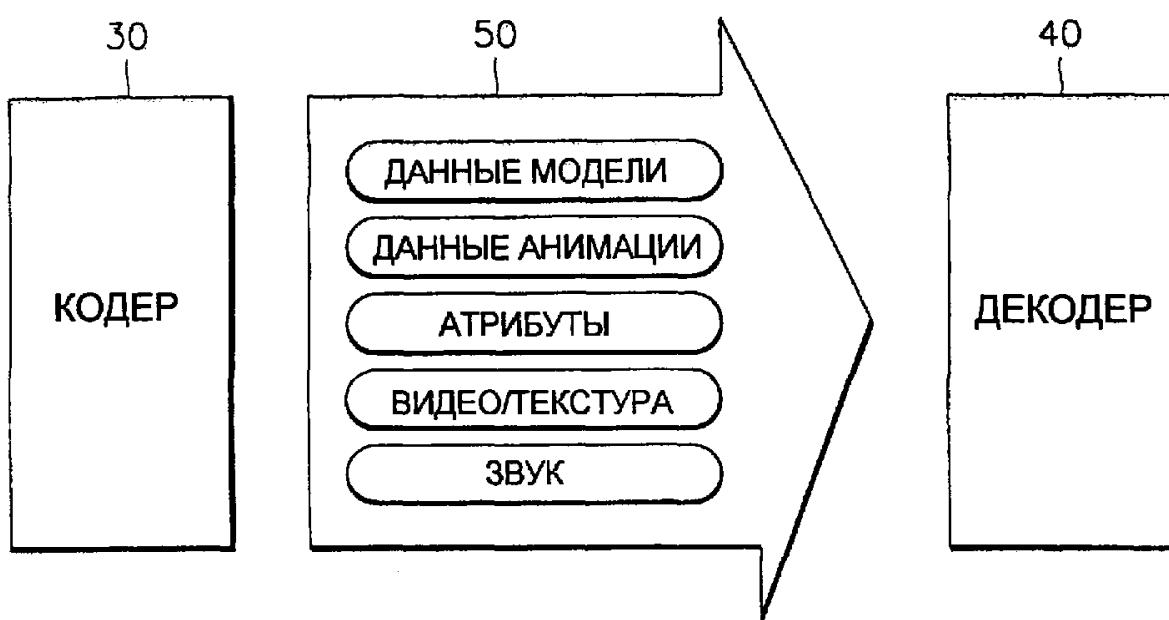
Фиг.1

R U ? 2 3 6 7 5 1 C 2

Скалярный интерполятор {  
 событие, заключающееся в разделении  
 плавающего набора SF  
 плавающий ключ [] открытого поля MF  
 плавающее ключевое значение [] открытого поля MF  
 событие в результате измененного плавающего  
 значения SF  
 }  
 ~

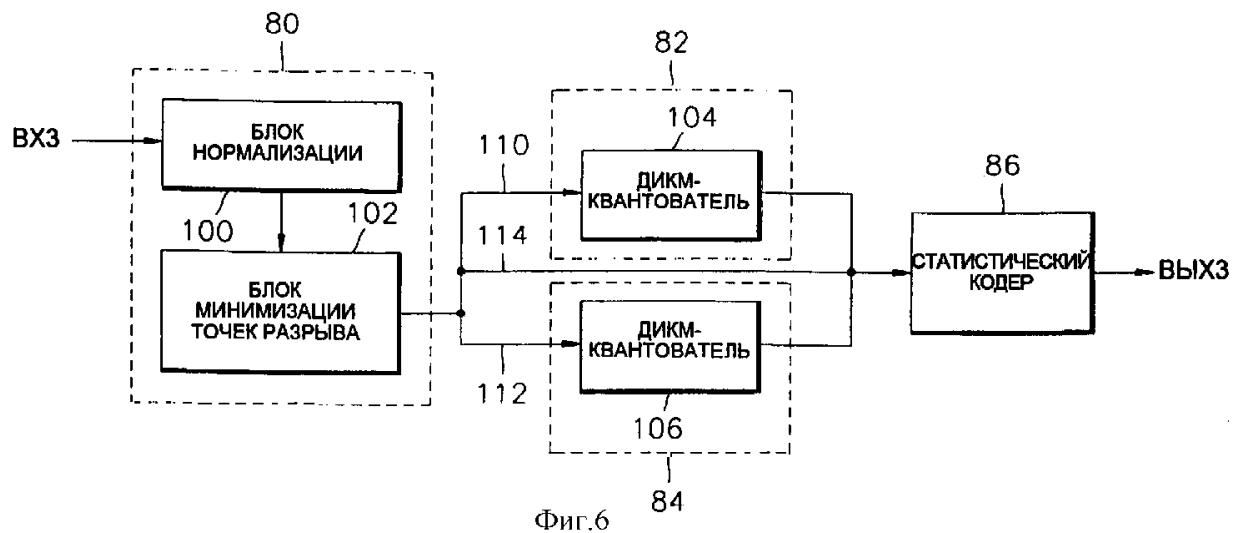
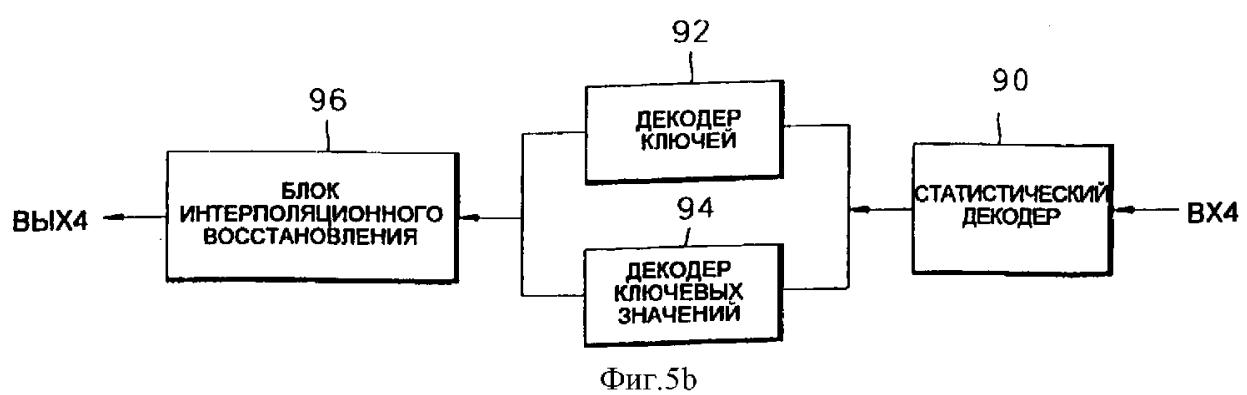
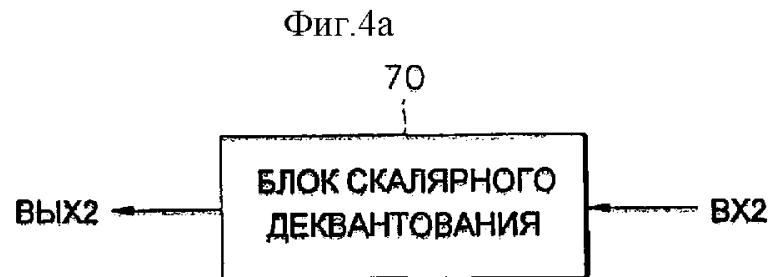
Фиг.2

R U ? 2 3 6 7 5 1 C 2

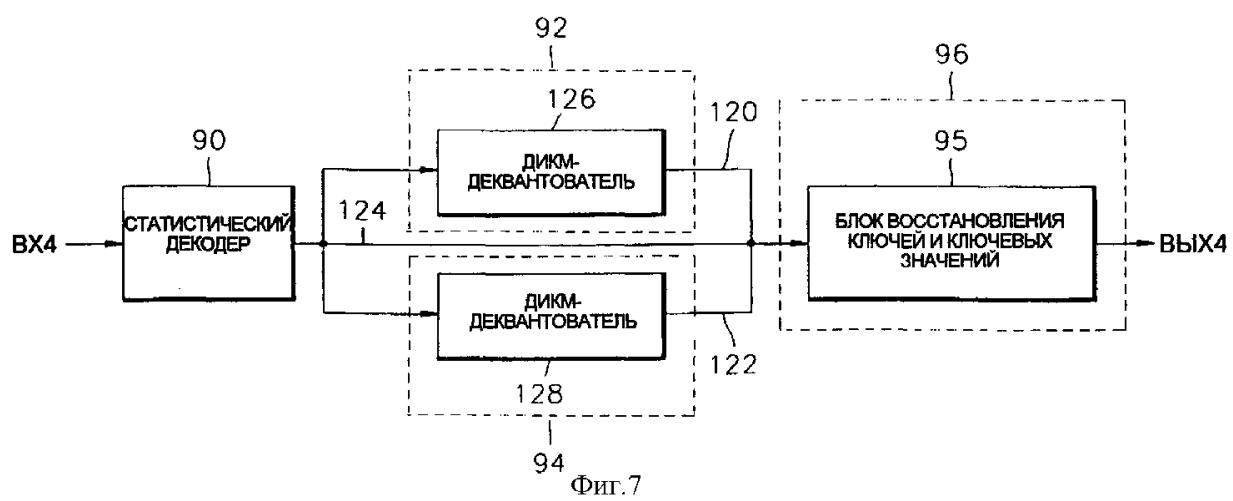


Фиг.3

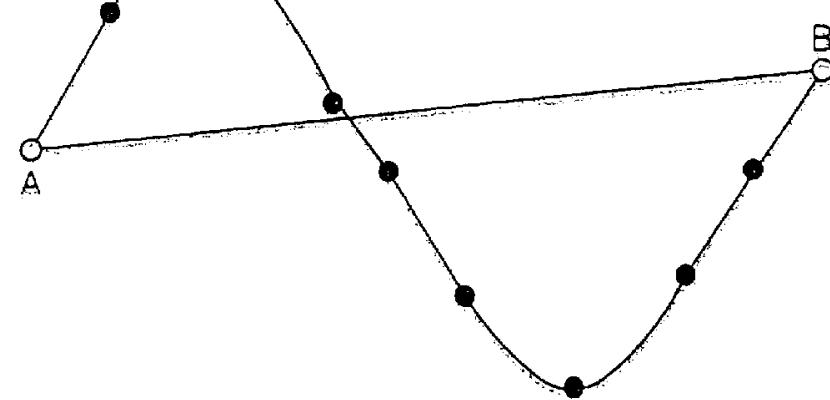
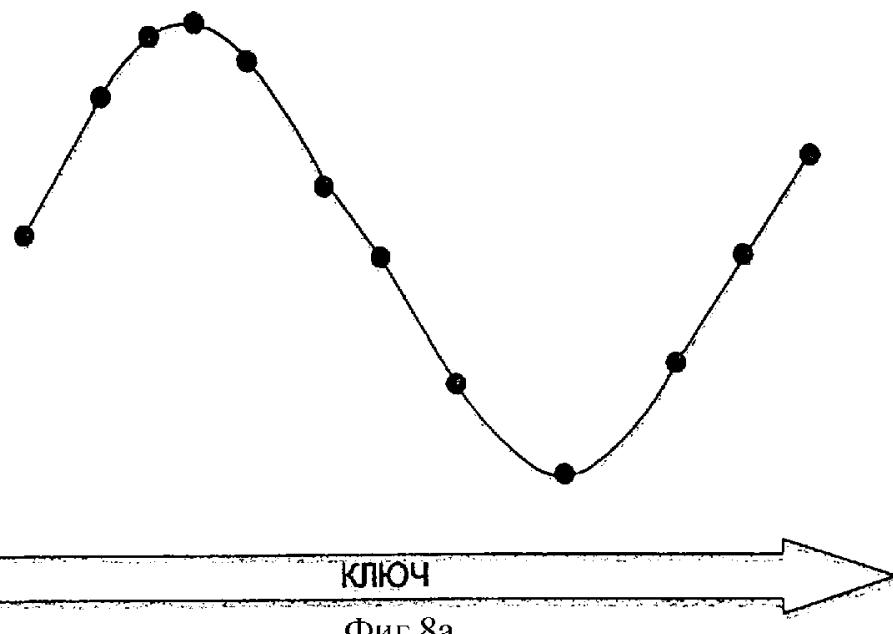
R U ? 2 3 6 7 5 1 C 2



R U 2 2 3 6 7 5 1 C 2



Фиг.7

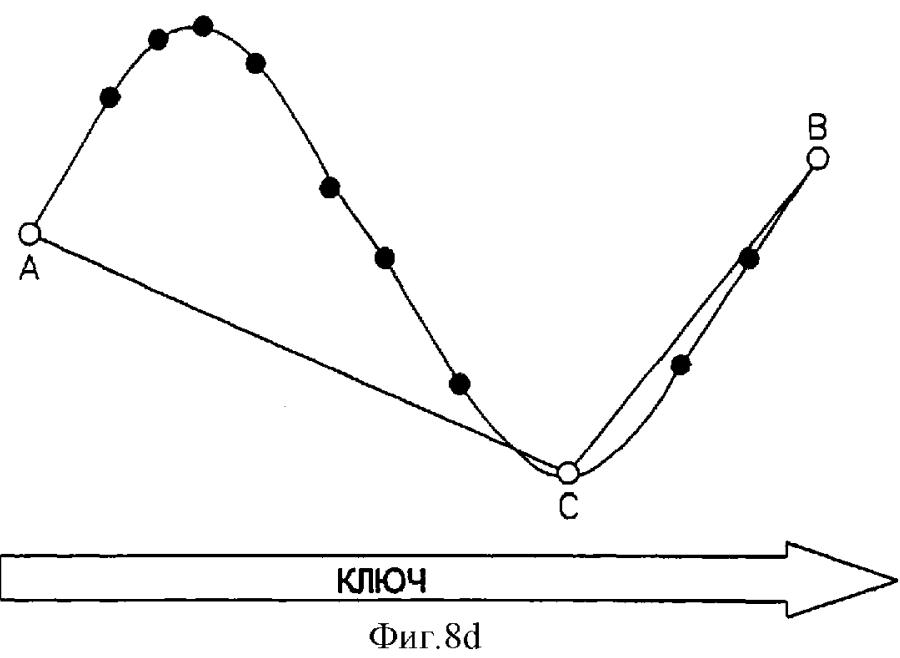
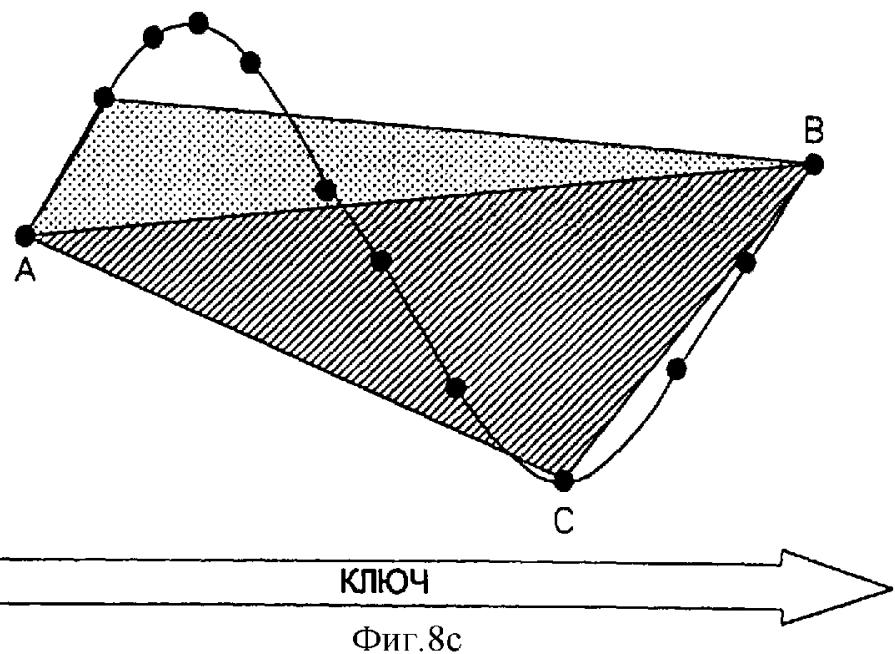


Ключ

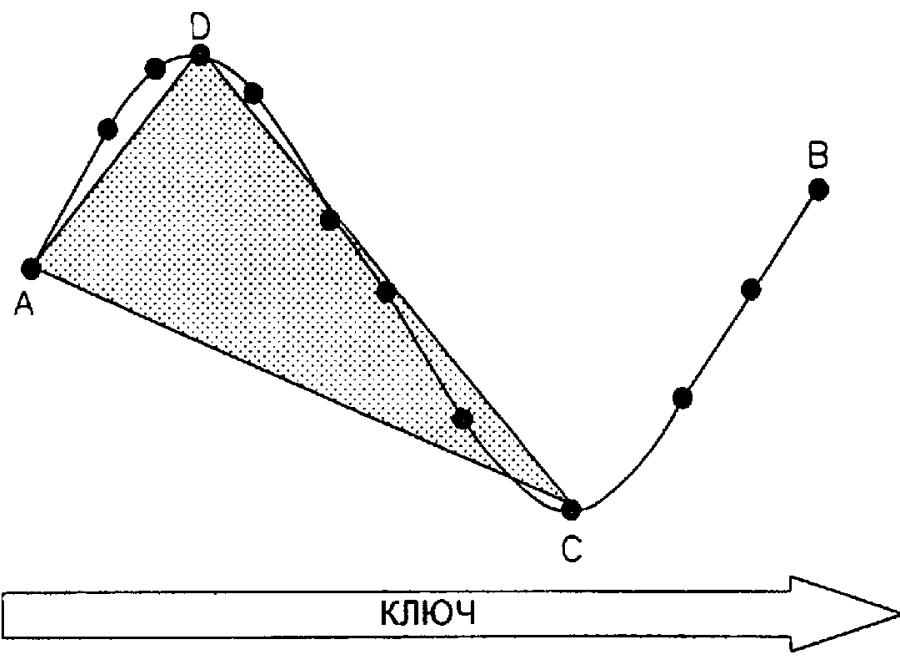
Фиг.8б

R U 2 2 3 6 7 5 1 C 2

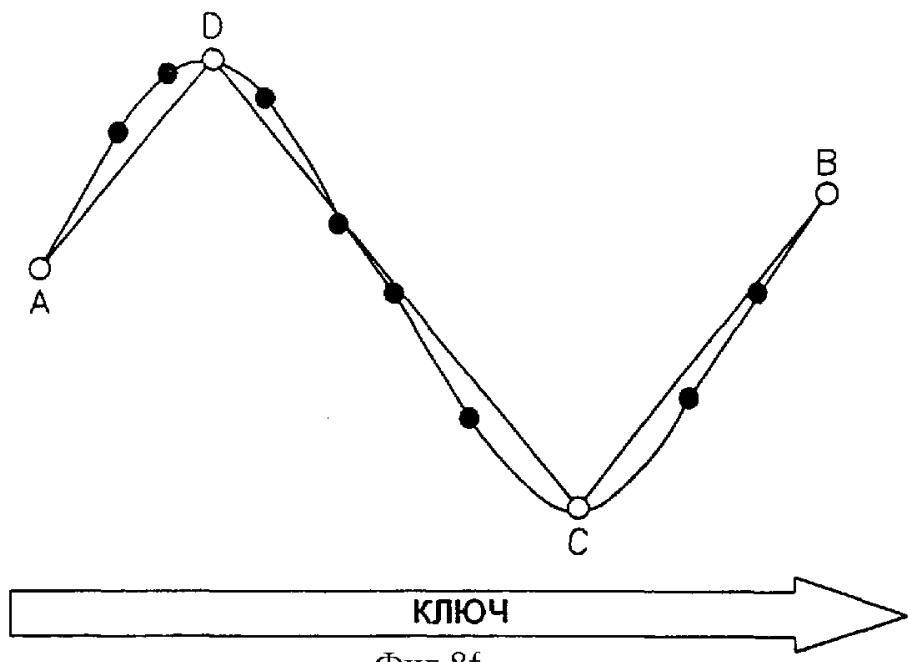
R U ? 2 3 6 7 5 1 C 2



R U 2 2 3 6 7 5 1 C 2



Фиг.8е

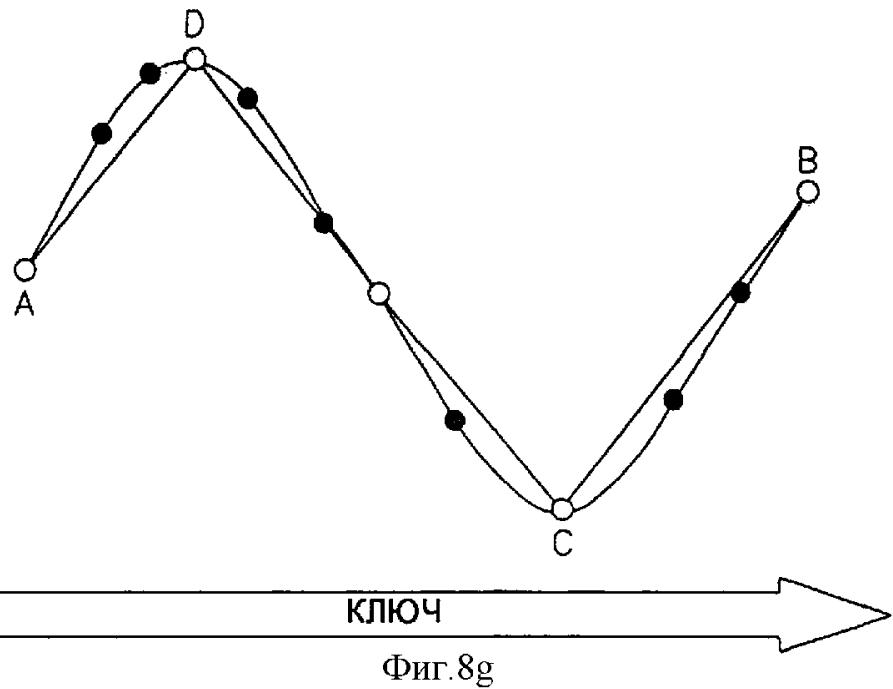


Фиг.8f

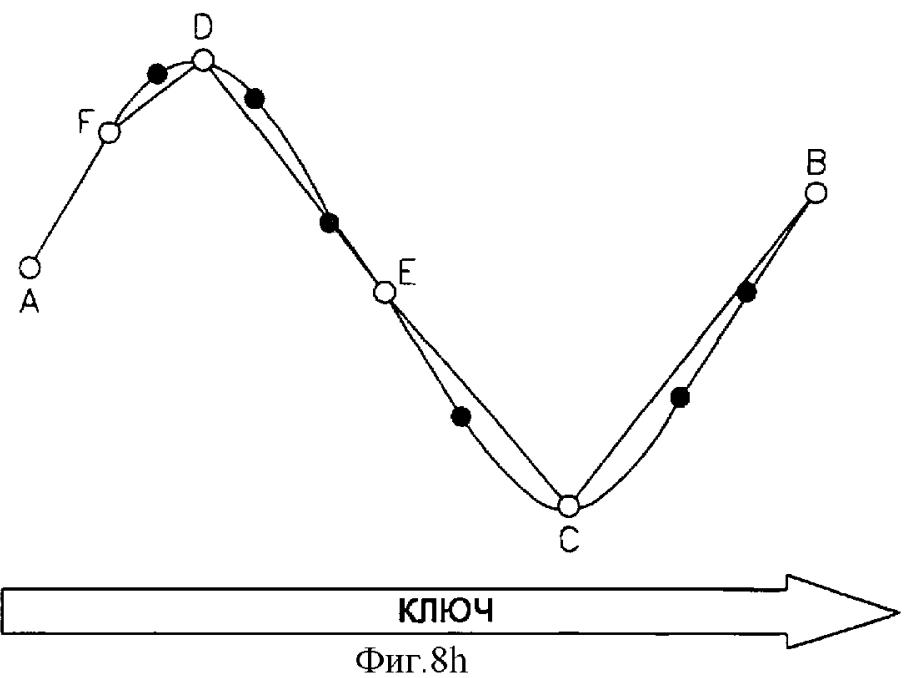
R U 2 2 3 6 7 5 1 C 2

R U ? 2 3 6 7 5 1 C 2

R U ? 2 3 6 7 5 1 C 2



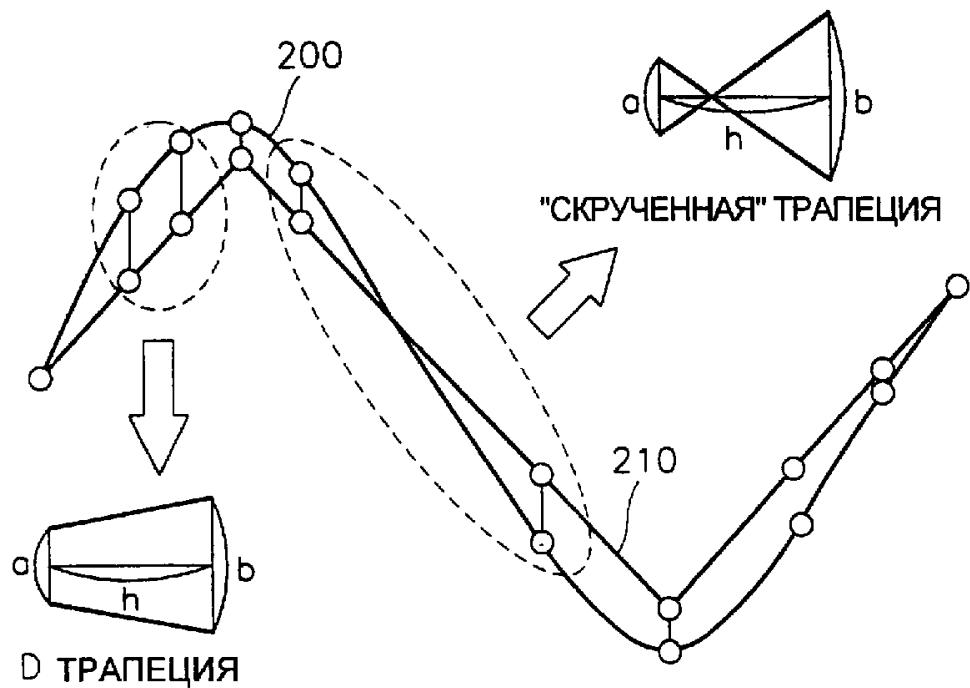
Фиг.8г



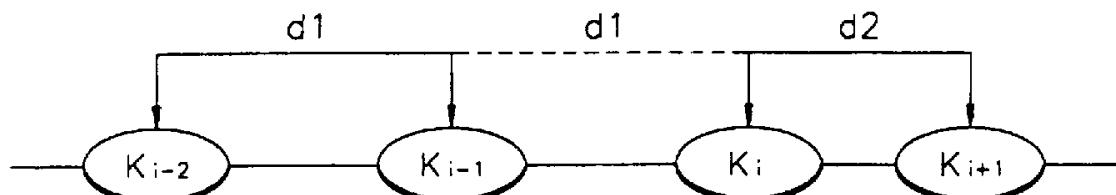
Фиг.8г

R U 2 2 3 6 7 5 1 C 2

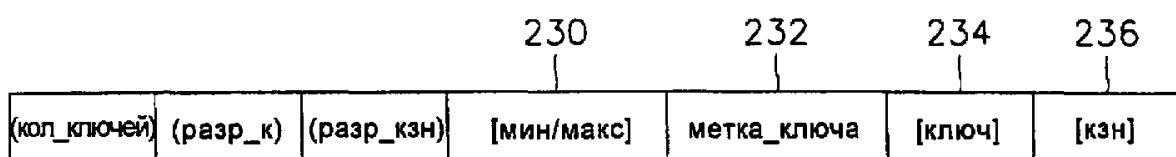
R U ? 2 3 6 7 5 1 C 2



Фиг.9



Фиг.10



Фиг.11а

мин <sub>x</sub>	макс <sub>x</sub>	мин <sub>y</sub>	макс <sub>y</sub>	мин <sub>z</sub>	макс <sub>z</sub>
------------------	-------------------	------------------	-------------------	------------------	-------------------

Фиг.11в

метка <sub>0</sub> _ключа	метка <sub>1</sub> _ключа	метка <sub>2</sub> _ключа	...	метка <sub>n-1</sub> _ключа
---------------------------	---------------------------	---------------------------	-----	-----------------------------

Фиг.11с

ключ <sub>0</sub>	ключ <sub>1</sub>	ключ <sub>2</sub>	...	ключ <sub>n-1</sub>
-------------------	-------------------	-------------------	-----	---------------------

Фиг.11д

кэн_X <sub>0</sub>	...	кэн_X <sub>p</sub>	кэн_Y <sub>0</sub>	...	кэн_Y <sub>q</sub>	кэн_Z <sub>0</sub>	...	кэн_Z <sub>r</sub>
--------------------	-----	--------------------	--------------------	-----	--------------------	--------------------	-----	--------------------

Фиг.11e

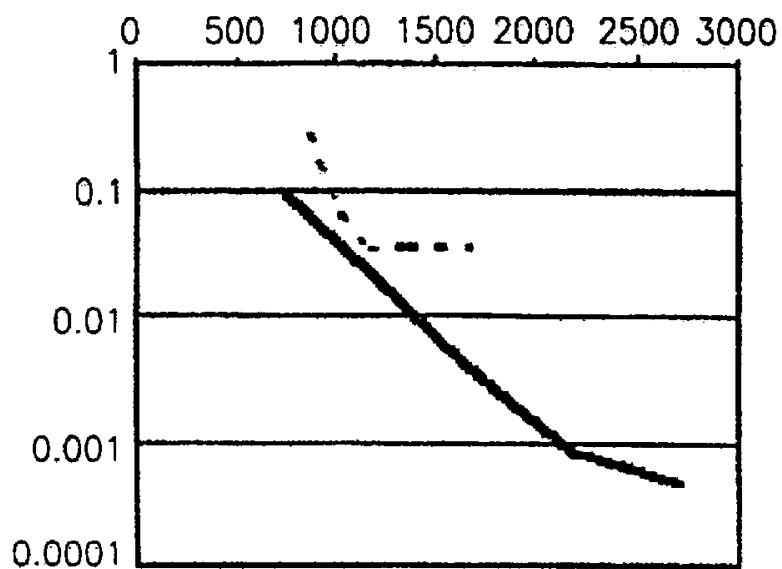
Сжатый позиционный интерполятор ()	Кол. битов	Мнемоника
{		
начало () декодирования AAC;	16	unimsbf
кол_ключей	16	unimsbf
разр_к	16	unimsbf
разр_кэн	16	unimsbf
мин_x		unimsbf
мин_y		unimsbf
мин_z		unimsbf
макс_x		unimsbf
макс_y		unimsbf
макс_z		unimsbf
для (i=0; i<кол_ключей; i++)		
{		
декодировать AAC (метка[i]_ключа, Контекст_метки_ключа);		vlclbf
}		
для (i=0; i<кол_ключей; i++)		
{		
декодировать AAC (метка[i]_ключа, Контекст_ключа);		vlclbf
}		
дешифровать_метку_ключа (метка_ключа);		vlclbf
для (i=0; i<кол_кэн_x; i++)		
{		
декодировать AAC(кэн[i], Контекст_кэн);		vlclbf
Обр_квантовать (кэн[i]);		
}		
для (i=0; i<кол_кэн_y; i++)		
{		
декодировать AAC(кэн[i], Контекст_кэн);		
Обр_квантовать (кэн[i]);		
}		
для (i=0; i<кол_кэн_z; i++)		
{		
декодировать AAC(кэн[i], Контекст_кэн);		vlclbf
Обр_квантовать (кэн[i]);		
}		
Закончить() декодирование AAC;		
}		

Фиг.12

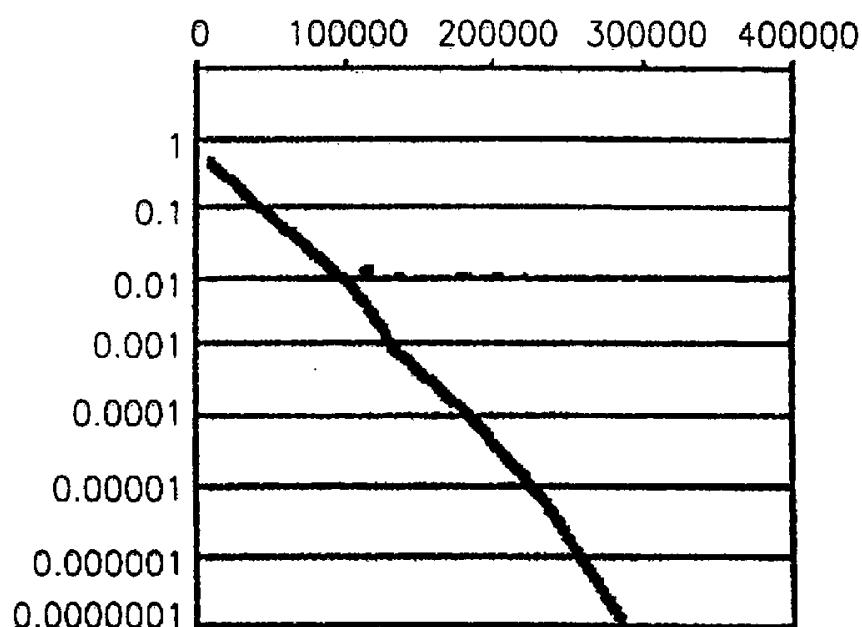
R U  
2 2 3 6 7 5 1 C 2

R U ? 2 3 6 7 5 1 C 2

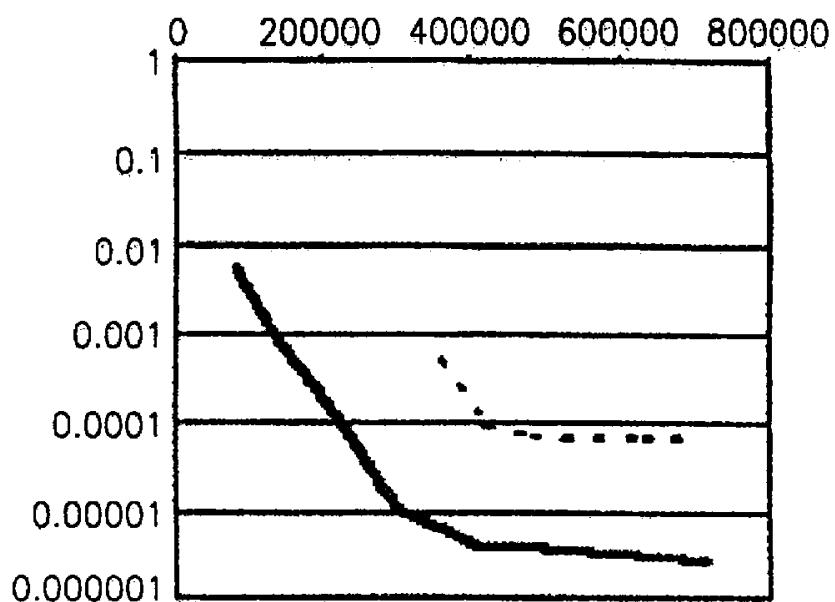
R U ? 2 3 6 7 5 1 C 2



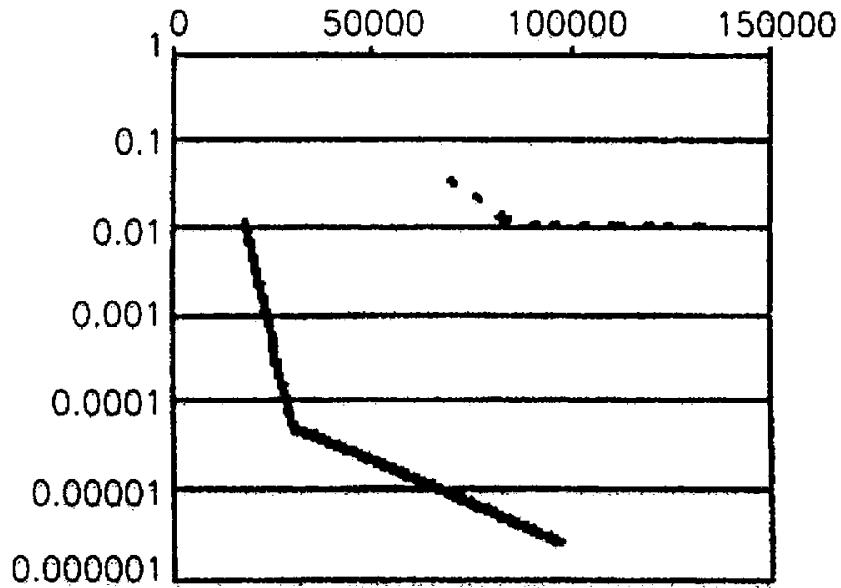
Фиг.13а



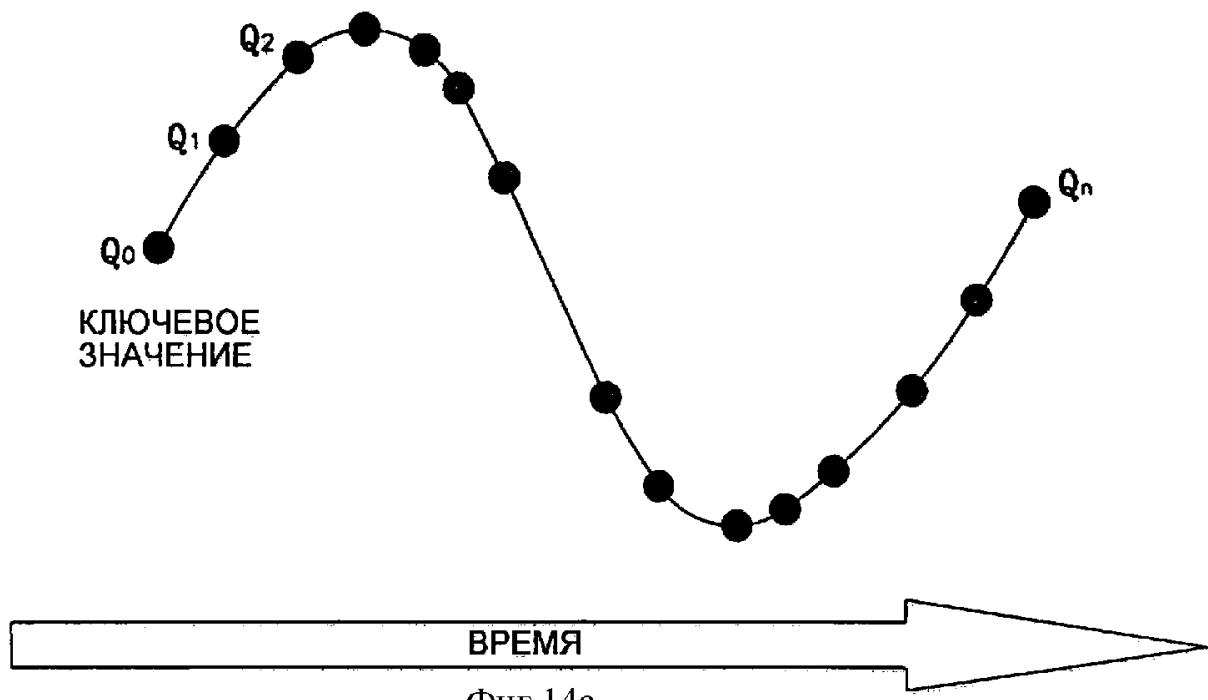
Фиг.13в



Фиг.13с



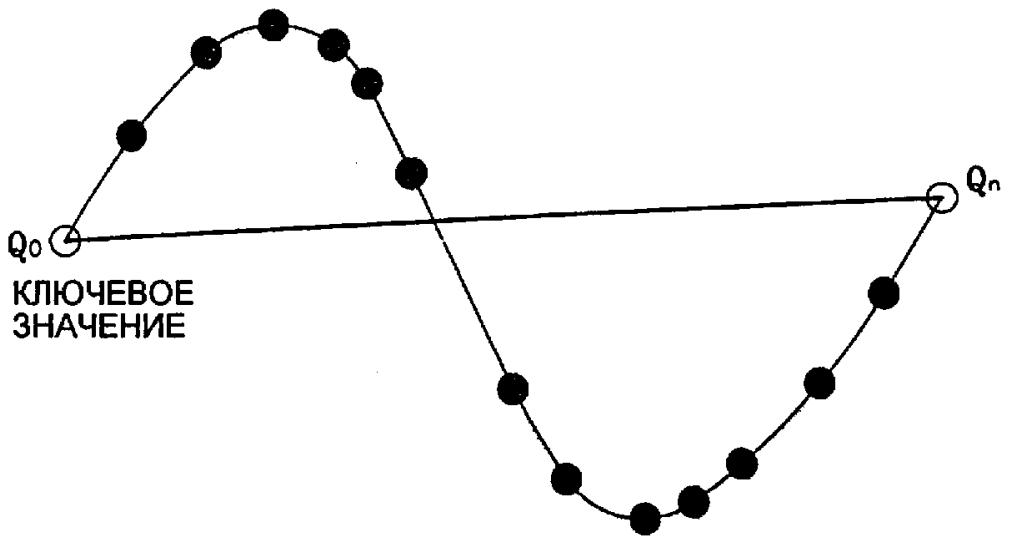
Фиг.13д



Фиг.14а

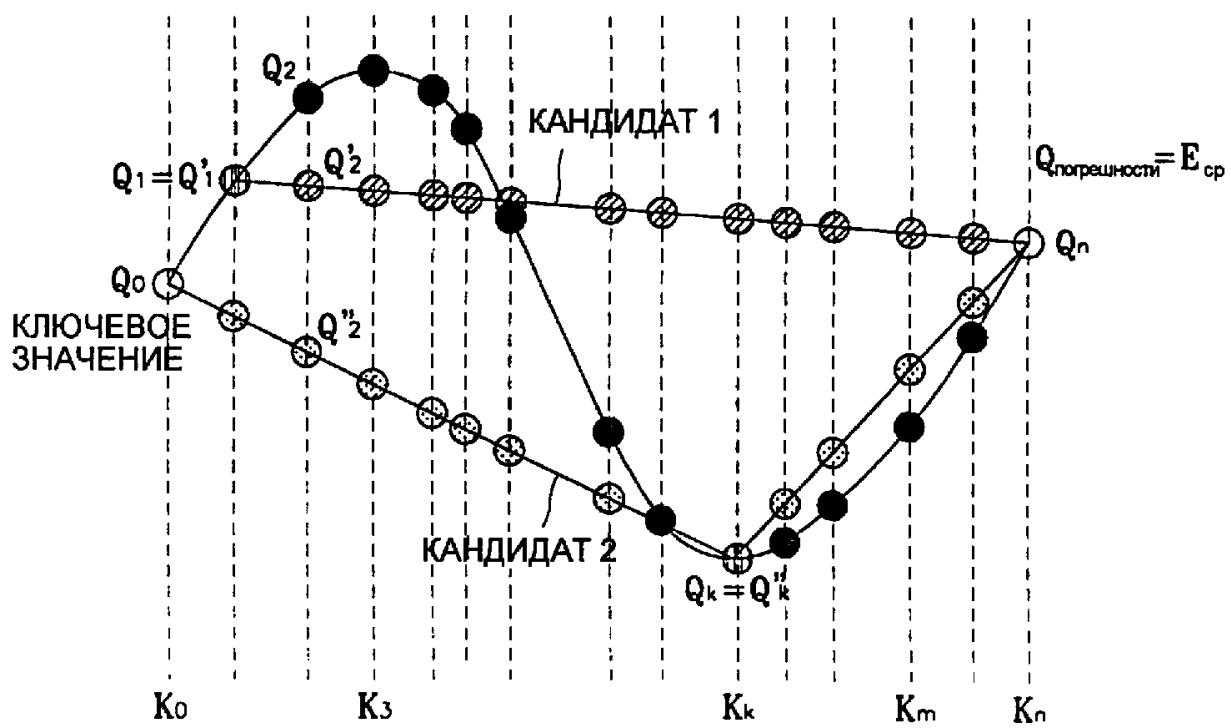
R U 2 2 3 6 7 5 1 C 2

R U ? 2 3 6 7 5 1 C 2



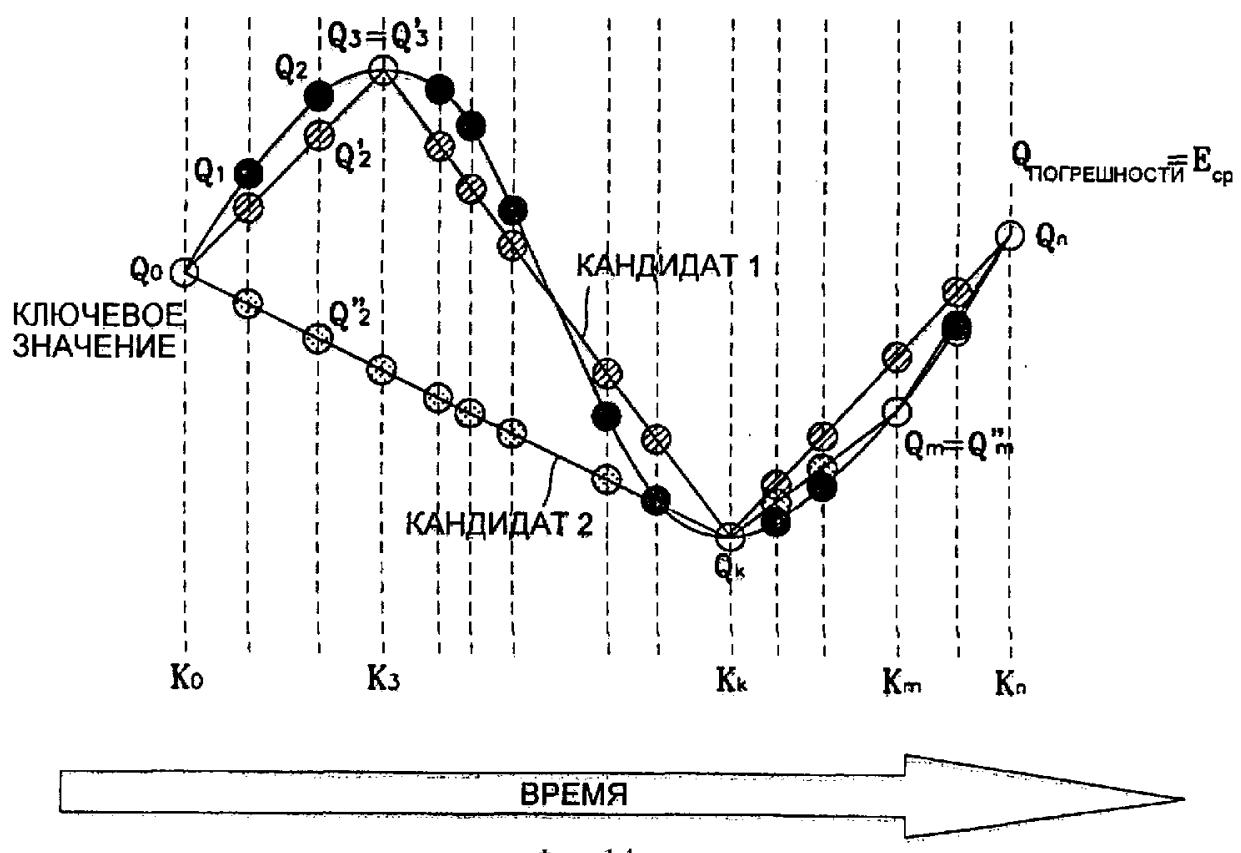
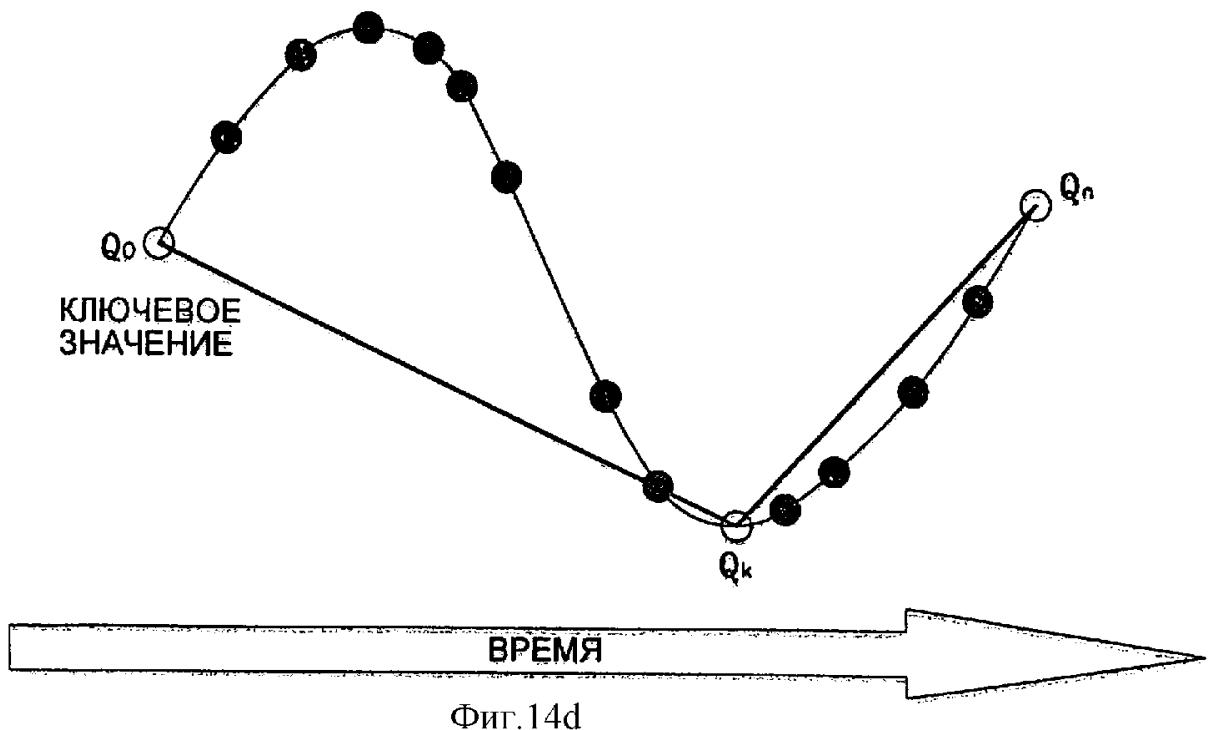
Фиг. 14в

R U ? 2 3 6 7 5 1 C 2

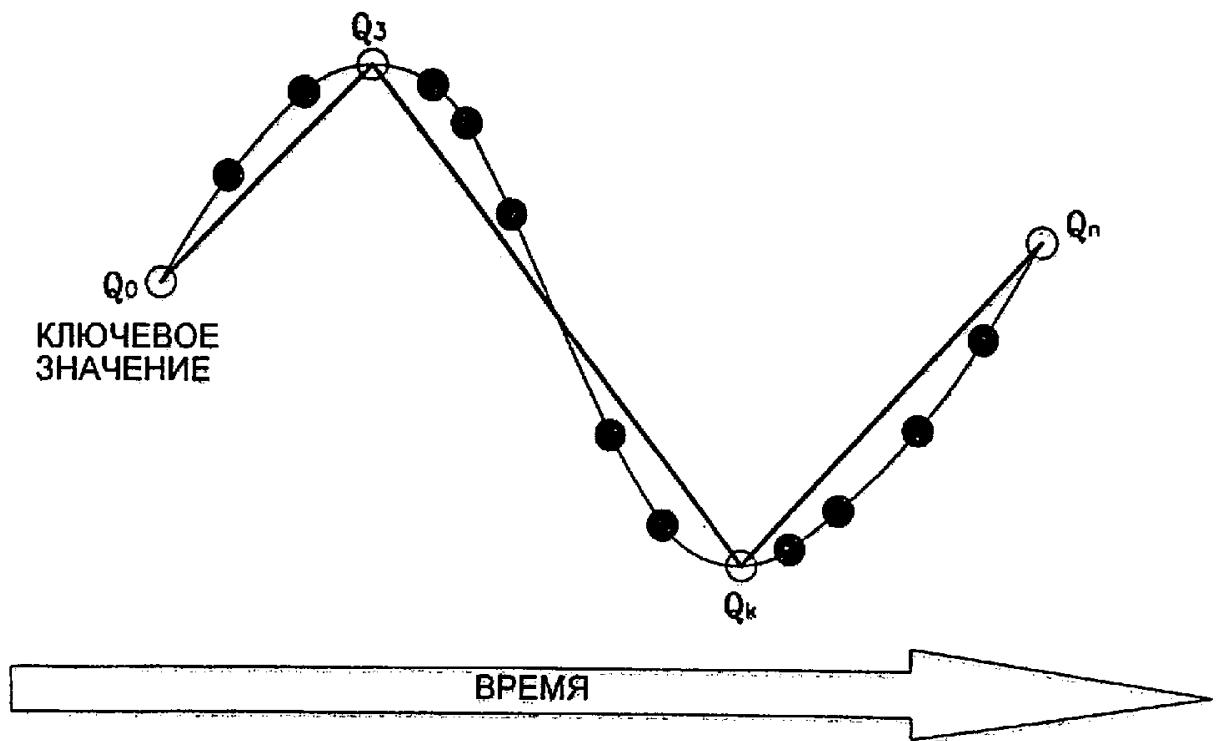


Фиг. 14с

R U 2 2 3 6 7 5 1 C 2



R U 2 2 3 6 7 5 1 C 2



Фиг. 14f

R U 2 2 3 6 7 5 1 C 2

R U ? 2 3 6 7 5 1 C 2