

(19) 日本国特許庁(JP)

(12) 特許公報(B2)

(11) 特許番号

特許第3613191号
(P3613191)

(45) 発行日 平成17年1月26日(2005.1.26)

(24) 登録日 平成16年11月5日(2004.11.5)

(51) Int. Cl.⁷

G10H 7/02
G10H 1/00

F I

G10H 7/00 521R
G10H 1/00 102A
G10H 7/00 521Q

請求項の数 5 (全 28 頁)

(21) 出願番号	特願2001-91187 (P2001-91187)	(73) 特許権者	000004075 ヤマハ株式会社 静岡県浜松市中沢町10番1号
(22) 出願日	平成13年3月27日(2001.3.27)	(74) 代理人	100077539 弁理士 飯塚 義仁
(65) 公開番号	特開2002-287760 (P2002-287760A)	(72) 発明者	田邑 元一 静岡県浜松市中沢町10番1号 ヤマハ株式会社内
(43) 公開日	平成14年10月4日(2002.10.4)	(72) 発明者	梅山 康之 静岡県浜松市中沢町10番1号 ヤマハ株式会社内
審査請求日	平成13年10月16日(2001.10.16)	(72) 発明者	増田 英之 静岡県浜松市中沢町10番1号 ヤマハ株式会社内

最終頁に続く

(54) 【発明の名称】 波形生成方法及び装置

(57) 【特許請求の範囲】

【請求項1】

調和成分の要素と調和外成分の要素とを含む奏法モジュールを順次指定するステップと、前記指定に従い、先行する奏法モジュールの少なくとも終端特性に関する終端情報と後続する奏法モジュールの少なくとも先端特性に関する先端情報とを、調和成分及び調和外成分の各要素毎に、取得するステップと、

取得した終端情報と先端情報に基づいて、前記先行する奏法モジュールと後続する奏法モジュールの少なくとも一方の特性を、前記調和成分及び調和外成分の各要素毎に、修正するステップであって、当該修正の方法は、前記先行する奏法モジュールの種類と前記後続する奏法モジュールの種類とに応じて定められているものと、

前記指定された奏法モジュールに対応する波形を前記修正に従う特性で合成するステップと

を具える波形生成方法。

【請求項2】

前記終端情報と先端情報は、時刻情報とレベル情報の少なくとも一方を含む請求項1に記載の波形生成方法。

【請求項3】

請求項1又は2に記載の波形生成方法をコンピュータに実行させるためのプログラム。

【請求項4】

請求項3に記載のプログラムを記憶したコンピュータ読み取り可能な記憶媒体。

【請求項5】

調和成分の要素と調和外成分の要素とを含む奏法モジュールを順次指定する手段と、前記指定に従い、先行する奏法モジュールの少なくとも終端特性に関する終端情報と後続する奏法モジュールの少なくとも先端特性に関する先端情報とを、調和成分及び調和外成分の各要素毎に、取得する手段と、取得した終端情報と先端情報に基づいて、前記先行する奏法モジュールと後続する奏法モジュールの少なくとも一方の特性を、前記調和成分及び調和外成分の各要素毎に、修正する手段であって、当該修正の方法は、前記先行する奏法モジュールの種類と前記後続する奏法モジュールの種類とに応じて定められているものと、前記指定された奏法モジュールに対応する波形を前記修正に従う特性で合成する手段とを具える波形生成装置。

10

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

この発明は、波形メモリ等からの波形データの読み出し等に基づき、楽音あるいは音声若しくはその他任意の音の波形を生成する方法及び装置に関し、特に、自然楽器固有の各種奏法若しくはアーティキュレーションによる音色変化を忠実に表現した波形を生成することができるものに関する。この発明は、電子楽器は勿論のこと、自動演奏装置、コンピュータ、電子ゲーム装置その他のマルチメディア機器等、楽音あるいは音声若しくはその他任意の音を発生する機能を有するあらゆる分野の機器若しくは装置または方法において広範囲に適用できるものである。なお、この明細書において、楽音波形という場合、音楽的な音の波形に限るものではなく、音声あるいはその他任意の音の波形を含んでいてもよい意味合いで用いるものとする。

20

【0002】

【従来の技術】

波形メモリにおいて、PCM（パルス符号変調）あるいはDPCM（差分PCM）又はADPCM（適応差分PCM）等の任意の符号化方式で符号化した波形データ（つまり波形サンプルデータ）を記憶しておき、これを所望の音楽ピッチに対応して読み出すことにより、楽音波形を形成するようにした、いわゆる「波形メモリ読み出し」技術は既に公知であり、また、様々なタイプの「波形メモリ読み出し方式」技術が知られている。従来知られた「波形メモリ読み出し方式」技術のほとんどは、発音開始から終了までの1つの音の波形を発生するためのものである。一例として、発音開始から終了までの1音の全波形の波形データを記憶する方式がある。また、別の例として、変化の複雑なアタック部などについてはその全波形の波形データを記憶し、変化のあまりないサステイン部などについては所定のループ波形を記憶する方式がある。なお、本明細書において、「ループ波形」とは繰り返し読み出し（ループ読み出し）される波形という意味で用いるものとする。

30

【0003】

ところで、従来の発音開始から終了までの1音の全波形の波形データを記憶する方式やアタック部などの波形の一部において全波形の波形データを記憶する方式の「波形メモリ読み出し方式」技術においては、各種奏法（若しくはアーティキュレーション）に対応する様々な波形データを多数記憶しておかなければならず、この多数の波形データを記憶するために大きな記憶容量が必要であった。

40

また、上述の全波形の波形データを記憶する方式では、自然楽器固有の各種奏法（若しくはアーティキュレーション）による音色変化を忠実に表現することが可能であるが、記憶した波形データの通りしか楽音を再生することができないので、制御性に乏しく、また、編集性にも乏しかった。例えば、所望の奏法（若しくはアーティキュレーション）に対応する波形データを演奏データに応じた時間軸制御等の特性制御を行うことが非常に困難であった。

【0004】

【発明が解決しようとする課題】

50

これらの点に鑑みて、自然楽器固有の各種奏法（すなわちアーティキュレーション）のリアルな再現とその制御を容易にした技術が、S A E M (S o u n d A r t i c u l a t i o n E l e m e n t M o d e l i n g) 技術と称して、特開2000-122665号公報その他に示されている。そのようなS A E M技術においては、複数の奏法波形モジュールを時系列的に組み合わせて接続することで一連の音波形を作成する場合に、不自然さを伴うことなく奏法波形モジュール同士の接続を行うことが望まれる。

本発明は上述の点に鑑みてなされたもので、様々な奏法（若しくはアーティキュレーション）に対応する高品質な波形データを容易かつ簡便にまた制御性豊かに生成することのできる波形生成方法及び装置を提供しようとするものであり、更には、その場合に、不自然さを伴うことなく奏法モジュール同士の接続を行うことができるようにした波形生成方法及び装置を提供しようとするものである。

10

【0005】

【課題を解決するための手段】

本発明に係る波形生成方法は、調和成分の要素と調和外成分の要素とを含む奏法モジュールを順次指定するステップと、前記指定に従い、先行する奏法モジュールの少なくとも終端特性に関する終端情報と後続する奏法モジュールの少なくとも先端特性に関する先端情報とを、調和成分及び調和外成分の各要素毎に、取得するステップと、取得した終端情報と先端情報に基づいて、前記先行する奏法モジュールと後続する奏法モジュールの少なくとも一方の特性を、前記調和成分及び調和外成分の各要素毎に、修正するステップとであって、当該修正の方法は、前記先行する奏法モジュールの種類と前記後続する奏法モジュールの種類とに応じて定められているものと、前記指定された奏法モジュールに対応する波形を前記修正に従う特性で合成するステップとを具える。

20

【0006】

本発明によれば、先行する奏法モジュールに対応する波形を実際に合成する前に、該先行する奏法モジュールの少なくとも終端特性に関する終端情報とそれに後続する奏法モジュールの少なくとも先端特性に関する先端情報とを取得する（リハーサルによる取得）。こうしてリハーサルとして取得した先行する奏法モジュールの少なくとも終端情報と後続する奏法モジュールの少なくとも先端情報の関係から両者が滑らかに接続されるように、少なくとも一方の特性を、該奏法モジュールに含まれる調和成分及び調和外成分の各要素毎に修正し、修正した特性をパラメータ若しくは制御データとして保持する。このように修正された特性に従い、奏法モジュールに対応する波形を実際に合成する。従って、相前後する奏法モジュールに基づく奏法波形同士が円滑に、かつ、調和成分及び調和外成分の各要素毎に独立に、接続されるようにすることができる。

30

【0007】

本発明は、方法の発明として構成し実施することができるのみならず、装置の発明として構成し実施することができる。また、本発明は、コンピュータまたはDSP等のプロセッサのプログラムの形態で実施することができるし、そのようなプログラムを記憶した記憶媒体の形態で実施することもできる。

【0008】

【発明の実施の形態】

以下、この発明の実施の形態を添付図面に従って詳細に説明する。

40

〔ハードウェア構成例〕

図1は、この発明の一実施例において利用可能な装置のハードウェア構成例を示すブロック図である。ここに示されたハードウェア構成例はコンピュータを用いて構成されており、そこにおいて、波形生成処理は、コンピュータがこの発明に係る波形生成処理を実現する所定のプログラム（ソフトウェア）を実行することにより実施される。勿論、この波形生成処理はコンピュータソフトウェアの形態に限らず、DSP（デジタル・シグナル・プロセッサ）によって処理されるマイクロプログラムの形態でも実施可能であり、また、この種のプログラムの形態に限らず、ディスクリート回路又は集積回路若しくは大規模集積回路等を含んで構成された専用ハードウェア装置の形態で実施してもよい。また、この

50

波形生成装置は、電子楽器あるいはカラオケ装置又は電子ゲーム装置又はその他のマルチメディア機器又はパーソナルコンピュータ等、任意の製品応用形態をとってよい。

【0009】

図1に示されたハードウェア構成例においては、コンピュータのメイン制御部としてのCPU101に対して、バスラインBL（データあるいはアドレスバス等）を介してリードオンリメモリ（ROM）102、ランダムアクセスメモリ（RAM）103、パネルスイッチ104、パネル表示器105、ドライブ106、波形取込部107、波形出力部108、ハードディスク109、通信インタフェース111がそれぞれ接続されている。CPU101は、奏法波形生成処理や通常の楽音合成（ソフトウェア音源）等の各種の処理を、所定のプログラムに基づいて実行する。これらのプログラムは、通信インタフェース111を介したネットワークあるいはドライブ106に装着されたCDやMO等の外部記憶メディア106A等から供給されてハードディスク109に記憶される。そして、実行時にハードディスク109からRAM103にロードされる。あるいは、ROM102にプログラムが記録されていてもよい。ROM102は、CPU101により実行あるいは参照される各種プログラムや各種データ等を格納するものである。ROM103は、演奏に関する各種情報やCPU101がプログラムを実行する際に発生する各種データを一時的に記憶するワーキングメモリとして、あるいは現在実行中のプログラムやそれに関連するデータを記憶するメモリとして使用される。RAM103の所定のアドレス領域がそれぞれの機能に割り当てられ、レジスタやフラグ、テーブル、メモリなどとして利用される。パネルスイッチ104は、各種の演奏条件や波形生成条件等の設定情報を入力したり、波形データ等のエディットや各種情報の入力等を行うための各種の操作子を含んで構成される。例えば、数値データ入力用のテンキーや文字データ入力用のキーボード、あるいはパネルスイッチ等である。この他にも音高、音色、効果等を選択・設定・制御するための各種操作子を含んでいてよい。パネル表示器105は、パネルスイッチ104により入力された各種情報やサンプリングされた波形データ等を表示する、例えば液晶表示パネル（LCD）やCRT等のディスプレイである。

【0010】

波形取込部107はA/D変換器を内蔵し、外部波形入力（例えば、マイクロフォンなどからの入力）されたアナログ楽音信号をデジタルデータに変換（サンプリング）してRAM103あるいはハードディスク109に該デジタル波形データとして取り込むものである。取り込んだ波形データを基にして奏法波形データベースの作成を行うことができるが、その詳細説明は省略する。また、CPU101による波形生成処理によって生成された波形データはバスラインBLを介して波形出力部108に与えられ、適宜バッファ記憶される。波形出力部108ではバッファ記憶された波形データを所定の出力サンプリング周波数にしたがって出力し、これをD/A変換してサウンドシステム108Aに送出する。こうして、波形出力部108から出力された楽音信号は、サウンドシステム108Aを介して発音される。ハードディスク109は、奏法に応じた波形を合成するためのデータ（後述する奏法テーブル、コードブック等のデータ）や通常の波形データ、各種音色パラメータ等からなる音色データなどのような演奏に関する複数種類のデータを記憶したり、前記CPU101が実行する各種プログラム等の制御に関するデータを記憶するデータベースとして機能する。

【0011】

ドライブ106は、奏法に応じた波形を合成するためのデータ（後述する奏法テーブル、コードブック等の各種データ）や通常の波形データ、多種多様な音色パラメータ等からなる音色データなどのような演奏に関する複数種類のデータを記憶したり、前記CPU101が実行する各種プログラム等の制御に関するデータを記憶したりするための着脱可能なディスク（外部記憶メディア106A）をドライブするものである。なお、前記ドライブ106によりドライブされる外部記憶メディア106Aはフロッピーディスク（FD）の他に、コンパクトディスク（CD-ROM・CD-RAM）、光磁気ディスク（MO）、あるいはDVD（Digital Versatile Diskの略）等の着脱自在な

10

20

30

40

50

様々な形態の記憶メディアであってよい。制御プログラムを記憶した外部記憶メディア 106Aをドライブ106にセットし、その内容(制御プログラム)をハードディスク109に落とさずに、RAM103に直接ロードしてもよい。なお、外部記憶メディア106Aを用いて、あるいはネットワークを介して制御プログラムを提供するやり方は、制御プログラムの追加やバージョンアップ等を容易に行うことができるので好都合である。

【0012】

通信インタフェース111は、例えばLANやインターネット、電話回線等の通信ネットワーク(図示せず)に接続されており、該通信ネットワークを介して、サーバコンピュータ等(図示せず)と接続され、当該サーバコンピュータ等から制御プログラムや各種データあるいは演奏情報などを波形生成装置側に取り込むためのものである。すなわち、ROM102やハードディスク109に制御プログラムや各種データが記憶されていない場合に、サーバコンピュータから制御プログラムや各種データをダウンロードするために用いられる。クライアントとなる波形生成装置は、通信インタフェース111を介してサーバコンピュータへと制御プログラムや各種データのダウンロードを要求するコマンドを送信する。サーバコンピュータは、このコマンドを受け、要求された制御プログラムやデータなどを通信インタフェース111を介してハードディスク109に蓄積することにより、ダウンロードが完了する。更に、MIDIインタフェースを含み、MIDIの演奏情報を受け取るようにしてもよいのは勿論である。また、音楽演奏用キーボードや演奏操作機器をバスラインBLに接続し、リアルタイム演奏によって演奏情報を供給するようにしてもよいのは言うまでもない。勿論、所望の音楽曲の演奏情報を記憶した外部記憶メディア106Aを使用して、演奏情報を供給するようにしてもよい。

【0013】

〔奏法モジュールの概略説明〕

上述のハードディスク109あるいはその他適宜の記憶媒体を用いて構成される奏法波形データベースにおいては、種々の奏法(すなわちアーティキュレーション)の要素に対応する波形を再生するための多数のモジュールデータ(これを「奏法モジュール」という)とそれに関連するデータ群を記憶している。1つの「奏法モジュール」とは、奏法波形合成システムにおいて1つのかたまりとして処理できる奏法波形の単位である。別の言い方をすると、「奏法モジュール」とは、1つのイベントとして処理できる奏法波形の単位である。例えば、種々有る奏法モジュールの中には、演奏音の奏法的特徴に応じて、アタックやボディあるいはリリース等の音の部分的区間に対応して定義されているものもあれば、また、スラーのような音と音のつながりの区間(ジョイント部)に対応して定義されているものもあり、ビブラートのような音の特殊演奏部分に対応して定義されているものもある。また、フレーズのように複数音符に対応して定義されているものもあってもよい。

【0014】

奏法モジュールは、奏法の特徴若しくは演奏の時間的単位又は区間等に基づき、大きくいくつかの種類に分類することができる。その例を示すと、次の5種類を挙げることができる。

- 1) 「ノーマル・エントランス」(略称NE)： (無音状態からの)音の立ち上がり部分(つまり「アタック」部分)を受け持つ奏法モジュール。
- 2) 「ノーマル・フィニッシュ」(略称NF)： (無音状態への)音の立ち下がり部分(つまり「リリース」部分)を受け持つ奏法モジュール。
- 3) 「ノーマル・ジョイント」(略称NJ)： 2つの音を(無音状態を経由せずに)接続する部分(つまり「ジョイント」部分)を受け持つ奏法モジュール。
- 4) 「ノーマル・ショート・ボディ」(略称NSB)： ビブラートのかからない、音の立ち上がり以降から立ち下がり以前までの部分(つまり「ボディ」部分)の短い部分を受け持つ奏法モジュール。
- 5) 「ビブラート・ロング・ボディ」(略称VLB)： ビブラートのかかった、音の立ち上がり以降から立ち下がり以前までの部分(つまり「ボディ」部分)を受け持つ奏法モジュール。

上記5種類の分類法は、明細書での説明のための一例にすぎず、他の分類法を採用してもよいし、更に多くの種類が存在してよい。また、奏法モジュールは、楽器種類等のオリジナル音源別にも分類されるのは勿論である。

【0015】

この実施例において、1つの奏法モジュールに対応する1つの奏法波形のデータは、そのままデータベースに記憶されているのではなく、複数の波形構成要素の集合からなるものとしてデータベースに記憶されている。この波形構成要素を、以下、「ベクトル」という。1つの奏法モジュールに対応するベクトルの種類には一例として下記のようなものがある。なお、調和成分及び調和外成分とは、対象たるオリジナル奏法波形をピッチ調和成分からなる波形ととそれ以外の残りの波形成分とに分離することで定義されるものである。 10

1) 調和成分の波形 (Timbre) ベクトル: 調和成分の波形構成要素のうち、ピッチと振幅をノーマライズした波形形状のみの特徴を抽出したもの。

2) 調和成分の振幅 (Amplitude) ベクトル: 調和成分の波形構成要素のうち、振幅エンベロープ特性を抽出したもの。

3) 調和成分のピッチ (Pitch) ベクトル: 調和成分の波形構成要素のうち、ピッチ特性を抽出したもの (例えば或る基準ピッチを基準にした時間的ピッチ変動特性を示すもの)。

4) 調和外成分の波形 (Timbre) ベクトル: 調和外成分の波形構成要素のうち、振幅をノーマライズした波形形状 (ノイズ的波形) のみの特徴を抽出したもの。

5) 調和外成分の振幅 (Amplitude) ベクトル: 調和外成分の波形構成要素のうち、振幅エンベロープ特性を抽出したもの。 20

上記のほかに、更に別の種類のベクトル (例えば波形の時間軸の進行を示す時間ベクトル) が含まれていてもよいが、便宜上、本実施例ではその説明を省略する。

【0016】

なお、奏法波形の合成に際しては、これらのベクトルデータに対して制御データに応じた加工処理を適宜施して時間軸上に配置することで、奏法波形の各構成要素に対応する波形若しくはエンベロープを演奏音の再生時間軸に沿ってそれぞれ構築し、このようにして時間軸上に配置された各ベクトルデータに基づいて所定の波形合成処理を行うことで、奏法波形を生成する。例えば、調和波形ベクトルに調和ピッチベクトルに応じたピッチ及びその時間変化特性を付与すると共に調和振幅ベクトルに応じた振幅及びその時間変化特性を付与することで調和成分の波形を合成し、調和外波形ベクトルに調和外振幅ベクトルに応じた振幅及びその時間変化特性を付与することで調和外成分の波形を合成し、調和成分の波形と調和外成分の波形とを加算合成することで、最終的な所定の奏法的特徴を示す演奏音波形つまり奏法波形を生成することができる。 30

【0017】

奏法モジュールのデータ形式の一例について図2を参照して説明する。一例として、1つの奏法モジュールは図2に示したような階層的なデータ構造によって特定される。第1階層では、1つの奏法モジュールは「奏法ID」(奏法識別情報)と「奏法パラメータ」の組み合わせによって特定される。「奏法ID」は、当該奏法モジュールを識別する情報であり、これによって個々の奏法モジュールを特定するものであり、データベースから必要なベクトルデータを読み出すための情報の1つとして機能しうる。「奏法ID」は、例えば「楽器情報」と「モジュールパーツ名」との組み合わせで分類されることが可能である。楽器情報とは、当該奏法モジュールが適用される楽器名(バイオリン、あるいはアルト・サクソ、あるいはピアノなど)を示す情報である。「モジュールパーツ名」とは、当該奏法モジュールの種類をその性格と共に示す情報(例えば「ノーマル・エントランス」や「バンド・エントランス」など)である。このような「楽器情報」と「モジュールパーツ名」の情報を、「奏法ID」の情報の中に入れてもよい。あるいは、「奏法ID」に付加してこれらの「楽器情報」と「モジュールパーツ名」の情報を持たせるようにし、或る「奏法ID」がどのような性格の奏法モジュールに関わるものであるかを、これらの「楽器情報」と「モジュールパーツ名」の情報からユーザが知得できるようにしてもよい 40 50

【0018】

「奏法パラメータ」は、当該奏法モジュールに係る波形の時間やレベルなどを制御するためのパラメータである。「奏法パラメータ」には、各奏法モジュールの性格に応じて適宜異なる1又は複数種類のパラメータが含まれていてよい。例えば、Violin [Normal Entrance] という楽器情報とモジュールパーツ名との組み合わせで特定可能な所定モジュールの場合には、Entrance直後の絶対音高やEntrance直後の音量などの種類の奏法パラメータが含まれていてよいし、Violin [Bend Up Entrance] という楽器情報とモジュールパーツ名との組み合わせで特定可能な別の所定モジュールの場合には、Bend Up Entrance終了時の絶対音高、Bend Up Entrance時のBend深さの初期値、Bend Up Entrance開始(ノートオンタイミング)~終了までの時間、Entrance直後の音量、あるいはBend Up Entrance中のデフォルトのカーブの時間的な伸縮などの種類の奏法パラメータが含まれていてよい。また、Violin [Normal Short Body] という楽器情報とモジュールパーツ名との組み合わせで特定可能な更に別の所定モジュールの場合には、当該モジュールの絶対音高、Normal Short Bodyの終了時刻-開始時刻、Normal Short Body開始時のダイナミクス、Normal Short Body終了時のダイナミクスなどの種類の奏法パラメータが含まれていてよい。この「奏法パラメータ」は、奏法IDと共にメモリ等によって予め記憶されていてもよいし、あるいはユーザの入力操作によって入力するようにしたり、あるいは既存のパラメータをユーザの操作によって適宜変更できるようになっていたりしてもよい。また、奏法波形の再生に際して、奏法IDのみが与えられ、奏法パラメータが与えられなかったような場合には、当該奏法IDにとって標準的な奏法パラメータを自動的に付加するようにしてもよい。また、処理の過程で、適宜のパラメータが自動的に生成されて付加されるようになっていてもよい。

10

20

【0019】

第2階層のデータは、奏法IDによって特定されるベクトルID等のデータからなる。奏法波形のデータベースにおいて「奏法テーブル」と称するテーブル若しくはメモリ部が含まれており、この奏法テーブルにおいて、各奏法IDに対応して、当該奏法IDに関わる奏法モジュールを構成するための複数の波形構成要素つまり上述の各ベクトルの識別情報(つまりベクトルID)が記憶されている。つまり、奏法IDに応じて奏法テーブルを読み出すことにより、これらのベクトルID等のデータを取得することができる。なお、奏法テーブルに記憶されている第2階層のデータは、上記ベクトルIDのデータに限らず、他の必要なデータが含まれていてよい。例えば、代表点値列データ(複数サンプル列の中の補正のための代表的サンプル点を指示するデータ)が含まれていてもよい。例えば、振幅ベクトルやピッチベクトルのようなエンベロープ波形形状のデータは、いくつかの代表点値のデータを持っていれば、その形状を再現可能であるので、エンベロープ波形形状のデータをテンプレートとしてそっくり記憶しておくことなく、代表点値列データとして記憶していればよい。以下、代表点値列データを「シェープ」(Shape)データともいう。更には、各成分要素毎のベクトルデータ(波形要素、ピッチ要素(ピッチエンベロープ)、振幅要素(振幅エンベロープ))の開始時間位置や終了時間位置などの情報等を奏法テーブルに記憶していてもよい。あるいは、これらの時間位置等のデータの全部又は一部は、前記「奏法パラメータ」に含まれていてもよい。換言すれば、「奏法パラメータ」の一部がベクトルIDと共に奏法テーブルに記憶されていてもよい。

30

40

【0020】

第3階層のデータは、各ベクトルIDによって特定される個々のベクトルデータからなる。奏法波形のデータベースにおいて「コードブック」と称するメモリ部が含まれており、このコードブックにおいて、各ベクトルIDに対応して、当該ベクトルIDに関わる具体的なベクトルデータ(例えばTimbre波形のテンプレートなど)が記憶されている。つまり、ベクトルIDに応じて「コードブック」から具体的なベクトルデータを読み出す

50

ことができる。

【0021】

奏法テーブルに記憶される「ベクトルID」及び「shape」（代表点値列）等のデータを含む、1つの奏法モジュールのための、具体的な各種データの一例を説明すると、次の通りである。

データ1：奏法モジュールのサンプル長。

データ2：ノートオンタイミングの位置。

データ3：調和成分の振幅（Amplitude）要素のベクトルIDと代表点値列。

データ4：調和成分のピッチ（Pitch）要素のベクトルIDと代表点値列。

データ5：調和成分の波形（Timbre）要素のベクトルID。

10

データ6：調和外成分の振幅（Amplitude）要素のベクトルIDと代表点値列。

データ7：調和外成分の波形（Timbre）要素のベクトルID。

データ8：調和成分の波形（Timbre）要素の塊部の開始位置。

データ9：調和成分の波形（Timbre）要素の塊部の終了位置（調和成分の波形（Timbre）要素のループ部の開始位置）。

データ10：調和外成分の波形（Timbre）要素の塊部の開始位置。

データ11：調和外成分の波形（Timbre）要素の塊部の終了位置（調和外成分の波形（Timbre）要素のループ部の開始位置）。

データ12：調和外成分の波形（Timbre）要素のループ部の終了位置。

【0022】

20

上記データ1～12について、図3を参照して説明する。

図3は、当該奏法モジュールに対応する実波形区間を構成する各成分及び要素の一例を模式的に示す図であり、上から当該区間における調和成分の振幅（Amplitude）要素、調和成分のピッチ（Pitch）要素、調和成分の波形（Timbre）要素、調和外成分の振幅（Amplitude）要素、調和外成分の波形（Timbre）要素の一例を示す。なお、図に示している数字は上記各データの番号に対応するように付してある。

【0023】

1は、当該奏法モジュールに該当する波形のサンプル長（波形区間長）である。例えば、当該奏法モジュールの基となったオリジナル波形データの全体の時間長さに対応している。2はノートオンタイミングの位置であり、当該奏法モジュールのどの時間位置にも可変に設定することが可能である。原則的には、このノートオンタイミングの位置から当該波形に従った演奏音の発音が始まるが、ベンドアタックなどの奏法によってはノートオンタイミングよりも波形成分の立ち上がり開始時点が先行する場合がある。例えば、バイオリンでは実際に音が出る前から弓による弦の擦りが始められているので、そのような発音開始前の奏法波形の出始めを正確にシミュレートすることに適している。3は、コードブックに記憶された調和成分の振幅（Amplitude）要素のベクトルデータを指し示すためのベクトルID及び代表点値列を示す（図において、黒く塗りつぶした正方形で示す2点が代表点を示す）。4は、調和成分のピッチ（Pitch）要素のベクトルデータを指し示すためのベクトルID及び代表点値列を示す。

30

40

【0024】

6は、調和外成分の振幅（Amplitude）要素のベクトルデータを指し示すためのベクトルID及び代表点値列を示す。代表点値列データはベクトルIDによって指示されるベクトルデータ（複数サンプル列からなる）を変更制御するためのデータであり、代表的サンプル点のいくつかを指示（特定）するものである。特定された代表的サンプル点に関してその時間位置（横軸）とレベル軸（縦軸）を変更若しくは補正することにより、他の残りのサンプル点も連動して変更し、もってベクトルの形状を変更する。例えば、そのサンプル数より少ない数の分散的サンプルを示すデータであるが、勿論これに限らず、代表点値列データはサンプルとサンプルの間の中間位置のデータであってもよいし、あるいは所定の範囲（連続的な複数サンプル）にわたるデータであってもよい。また、サンプル

50

値そのものでなく、差分や乗数等のデータであってもよい。この代表点を横軸及びノ又は縦軸（時間軸）に移動することによって、各ベクトルデータの形状を変えることができる。つまり、エンベロープ波形の形状を変えることができる。5は、調和成分の波形（Timbre）要素のベクトルデータを指し示すためのベクトルIDである。7は、調和外成分の波形（Timbre）要素のベクトルデータを指し示すためのベクトルIDである。8は、調和成分の波形（Timbre）要素の波形の塊部の開始位置である。9は、調和成分の波形（Timbre）要素の波形の塊部の終了位置（あるいは、調和成分の波形（Timbre）要素の波形のループ部の開始位置）である。すなわち、8から開始する三角形は特徴のある波形形状が連続的に記憶されているノンループ波形の部分を示し、その後続く9から開始する長方形は繰り返し読み出しすることのできるループ波形の部分を示す。ノンループ波形は、奏法（若しくはアーティキュレーション）等の特徴を有する高品質な波形である。ループ波形は、1周期または適当な複数周期分の波形からなる比較的単調な音部分の単位波形である。

【0025】

10は、調和外成分の波形（Timbre）要素の波形の塊部の開始位置である。11は、調和外成分の波形（Timbre）要素の波形の塊部の終了位置（あるいは、調和外成分の波形（Timbre）要素の波形のループ部の開始位置）である。12は、調和外成分の波形（Timbre）要素の波形のループ部の終了位置である。上記データ3～データ7は各成分要素毎にコードブックに記憶されているベクトルデータを指し示すための識別情報のデータであり、上記データ2及びデータ8～データ12はベクトルデータから元の（分離前の）波形を組み立てるための時間情報のデータである。このように、奏法モジュールのデータはベクトルデータを指し示すためのデータと時間情報のデータとから構成される。このような奏法テーブルに記憶されている奏法モジュールのデータを使用することにより、コードブックに記憶されている波形の素材（ベクトルデータ）を使って、波形を自由に組み立てることができることになる。つまり、奏法モジュールは、奏法（若しくはアーティキュレーション）に応じて生成する波形の挙動を表すデータである。なお、奏法モジュールのデータの種類や数は各奏法モジュール毎に異なっていてよい。また、上述したデータ以外にも他の情報等を具えていてよい。例えば、波形の時間軸を伸長/圧縮制御するためのデータなどを持っていてもよい。

【0026】

また、上述の例では説明を理解しやすくするために、1つの奏法モジュールが調和成分の各要素（波形、ピッチ、振幅）及び調和外成分の各要素（波形、振幅）の全てを具備している例について説明したが、これに限らず、奏法モジュールが調和成分の各要素（波形、ピッチ、振幅）や調和外成分の各要素（波形、振幅）の1つからなっていてよいのは勿論である。例えば、奏法モジュールが調和成分の波形（Timbre）要素、調和成分のピッチ（Pitch）要素、調和成分の振幅（Amplitude）要素、調和外成分の波形（Timbre）要素、調和外成分の振幅（Amplitude）要素のいずれか1つの要素からなっていてよい。こうすると、各成分毎に奏法モジュールを自由に組み合わせ使用することができることになり好ましい。

【0027】

〔演奏データの一例の説明〕

この実施例においては、所望の曲の自動演奏データ（曲ファイル）の中に奏法波形を再生するための演奏イベントデータが含まれており、自動演奏シーケンスの進行に伴って読み出される演奏イベントデータに基づき奏法波形が生成される。自動演奏データ（曲ファイル）は、基本的に、SMF（スタンダードMIDIファイル）形式からなり、通常のMIDIデータとAEM（アーティキュレーション・エレメント・モデリング）演奏イベントデータ（換言すれば奏法イベントデータ）とが混在する演奏データからなる。例えば、1つの曲の自動演奏データが複数トラックの演奏データからなり、そのうちの1又は複数トラックがAEM演奏イベント（奏法イベント）を含むAEM演奏シーケンスのトラックであり、他のトラックが通常のMIDI演奏シーケンスのトラックである。また、1トラッ

10

20

30

40

50

ク内でMIDIデータとAEM演奏イベント(奏法イベント)データとが混在していてもよい。その場合は、AEM演奏イベント(奏法イベント)データは、基本的にMIDIフォーマットで記述され、MIDIチャンネルのいずれか1又は複数がAEM演奏データのために割り当てられる。また、1トラック全体がAEM演奏データのために割り当てられている場合でも、データは基本的にはMIDIフォーマットで記述されている。つまり、MIDIフォーマットの中で、AEM演奏イベント(奏法イベント)であることを示す識別子を付加しておけばよい。勿論、MIDI以外のデータフォーマットを使用してもよい。各トラックの演奏データは別々の演奏パートの演奏データを構成する。また、1つのトラックの演奏データ内に複数のMIDIチャンネルの演奏データが混在するので、1トラックの演奏データにおいても各MIDIチャンネル毎に別々の演奏パートの演奏データを構成しうる。例えば、或る1又は複数の演奏パートの演奏音が、AEM演奏データに基づく奏法波形合成によって、再生される。例えば、バイオリンのパートと、ピアノのパート、というように、複数の演奏パートでそれぞれAEM演奏データに基づく奏法波形合成を行うことができる。

10

【0028】

図4(a)は、1つの曲の自動演奏データの全体的構成例を示し、ヘッダと各トラック1, 2, 3, ……毎の演奏データ列からなる。図4(b)は、AEM演奏データを含むトラック(例えばトラック2)における演奏データ列の一例を示す。この演奏データ列の構成は、通常の演奏シーケンスデータにおいてよく知られているように、時間差データ(デュレーションデータ)とイベントデータとからなり、時間順にイベントデータを配列してなるものである。公知のように、時間差データ(デュレーションデータ)は、前のイベント時点から次のイベント時点までの時間差を示している。

20

1つの奏法イベントは、前記図2に示した第1階層のデータ、つまり、当該イベントに対応して再生すべき「奏法モジュール」を示す「奏法ID」と、それに関連する「奏法パラメータ」とを含む。前述のように、「奏法パラメータ」の全部又は一部がこの段階で含まれていなくてもよい。

【0029】

図4(b)の例では、「奏法イベント(1)」はアタック部(エントランス部)の「奏法モジュール」を指示する「奏法ID」を含んでおり、これと対になって、発音開始を指示する「ノートオンイベント」が配置されている。この「奏法イベント(1)」によって指示されるアタック部(エントランス部)の奏法波形の発音開始時点は、それと対で与えられる「ノートオンイベント」によって指定されるようになっている。アタック部の「奏法イベント(1)」とそれに対応する「ノートオンイベント」は一緒のイベントとして処理される。よって、アタック部(エントランス部)に対応する「奏法イベント(1)」の到来時刻は、アタック部(エントランス部)の奏法波形を生成する準備を開始すべきことを示しているだけであり、その発音開始時刻を指示するものではない。なお、後述するように、このアタック部(エントランス部)の奏法波形は、その発音が開始される以前から波形生成開始できるような仕様となっており、生成されたアタック部(エントランス部)の奏法波形の先頭からではなく、それと対を成すノートオンイベントの発生時点に対応する途中の時点からその発音を開始させることができるようになっている。これは、楽器の種類によっては演奏操作の始まり(例えばバイオリンの弓の擦り始め)では、その操作に応じた振動が人の耳に聞こえるほどの振動音としては発生されていないような状況をシミュレートすることに役立つ。また、そうでない場合でも、波形生成の自由度と制御性を高めるのに役立つ。

30

40

【0030】

更に図4(b)の例では、「奏法イベント(2)」はボディ部の「奏法モジュール」を指示する「奏法ID」を含み、「奏法イベント(3)」はジョイント部の「奏法モジュール」を指示する「奏法ID」を含み、「奏法イベント(4)」は別のボディ部の「奏法モジュール」を指示する「奏法ID」を含む。ジョイント部の「奏法モジュール」は、先行する音の発音を消音させることなく後続する音の発音へとつなげるときに(例えばタイヤス

50

ラーの奏法のときに)使用される「つなぎ」用の奏法波形である。よって、このジョイント部の「奏法イベント(3)」も、前記アタック部の奏法イベント(1)と同様に、「ノートオンイベント」(これは後続音の発音開始タイミングを示す)と対になって与えられるようになっており、図示のように、「奏法イベント(3)」の次に「ノートオンイベント」が発生する。ジョイント部の「奏法イベント(3)」とそれに対応する「ノートオンイベント」も一緒にイベントとして処理される。「奏法イベント(4)」は前記ジョイント部を介して接続される後続音のボディ部の奏法モジュールを指示するものである。「奏法イベント(5)」はリリース部(フィニッシュ部)の「奏法モジュール」を指示する「奏法ID」を含んでおり、これと対になって、消音開始(リリース)を指示する「ノートオフイベント」が配置されている。リリース部の「奏法イベント(5)」とそれに対応する「ノートオフイベント」も一緒にイベントとして処理される。ノートオンイベントの場合と同様に、リリース部(フィニッシュ部)の奏法波形は、その消音処理(ノートオフイベント)が開始される以前から波形生成開始できるような仕様となっており、生成されたリリース部(フィニッシュ部)の奏法波形の先頭からではなく、それと対を成すノートオフイベントの発生時点に対応する途中の時点からその消音処理を開始させることができるようになっており、これも上記と同じ理由であり、楽器の種類によって演奏操作の終わりの状況をシミュレートすることに役立つ、また、そうでない場合でも、波形生成の自由度と制御性を高めるのに役立つ。

【0031】

〔奏法波形生成処理の説明〕

図1に示す波形生成装置において、通常の楽音波形及び奏法波形の合成は、コンピュータが通常の音源プログラム及び本実施例に係る奏法波形生成プロセスを実現する所定のプログラム等を実行することにより実施される。この奏法波形生成プロセスを構成する大まかな処理ブロックは、図5に示すように、イージー・プレーヤー部20、奏法シーケンス部21、パート部22、奏法合成部23、波形合成部24からなる。図6は、この奏法波形生成プロセスを構成する各処理ブロックが受け持つ処理の時間関係を大まかに示す概略タイミングチャートである。図6において並列的に示されたブロック30, 31, 32, 33, 34は、奏法波形生成プロセスにおけるそれぞれの処理ブロック(イージー・プレーヤー部20、奏法シーケンス部21、パート部22、奏法合成部23、波形合成部24)が自己が受け持つ動作を実行する時間帯をおおまかに示すものである。図6において各ブロック30, 31, 32, 33, 34が並列的に図示されているのは、各ブロックの処理が並列的に行われることを意味する。

【0032】

(1) イージー・プレーヤー部20

イージー・プレーヤー部20は、再生しようとする任意の曲の自動演奏データ(曲ファイル)を記憶した記憶媒体から該自動演奏データ(曲ファイル)を読み込む機能、該再生しようとする曲等に関する入力操作子を介した各種の設定操作(移調量設定、音量調整、その他)及び指示操作(再生開始指示、再生停止指示、その他)を受け付ける機能、再生中の再生位置(時間)表示を含む各種表示を制御する機能、後述する必要情報補充機能、などを実行する。

図6における時間ブロック30がイージー・プレーヤー部20による処理が行われる時間帯を略示している。自動演奏の再生プログラムが起動されたときから終了するまでの間、イージー・プレーヤー部20が動作する。再生開始指示「PLAY」が与えられると、イージー・プレーヤー部20では、図6のブロック301で示すようなタイミングで、再生しようとする曲の自動演奏データ(曲ファイル)を読み込み、その解釈を行う。勿論、「PLAY」指示の以前に、所望の曲ファイルが選択された時点で、再生しようとする曲の自動演奏データ(曲ファイル)の読み込みが開始されてもよい。

【0033】

イージー・プレーヤー部20は、前述のとおり通常のMIDIデータとAEM演奏データとが複数トラック間で又は1トラック内で混在している自動演奏データを取り扱うことが

10

20

30

40

50

できるように構成されている。更に、イージー・プレーヤー部 20 では、読み込んだ自動演奏データ（曲ファイル）に含まれる A E M 演奏データに関して基本的な解釈を行い、各奏法イベントデータをタイムスタンプ付きの奏法イベント列オブジェクトとして再構成する。この基本的な解釈とは、例えば、各イベント間に存在する時間差データを累積してこれに基づき各イベント毎のタイムスタンプ（一連の曲中の絶対時間情報）を作成することなどを含む。再構成したタイムスタンプ付きの奏法イベント列オブジェクトは、奏法シーケンス部 21 に送付される（つまり奏法シーケンス部 21 が取り扱うメモリ内に書き込まれる）。この演奏データ解釈とこれに基づく奏法イベント列オブジェクトの再構成処理は、図 6 におけるブロック 301, 302 の時間帯が終了するまでに行われる。

【0034】

このように、イージー・プレーヤー部 20 は、基本的には、既存の自動演奏シーケンスデータと類似構成からなる（図 4 に示したような）A E M 演奏データを、奏法シーケンス部 21 で容易に取り扱うのに適したデータ列に変換する。なお、通常の M I D I 演奏データは、イージー・プレーヤー部 20 内に含まれる M I D I 用の公知の演奏シーケンサ（M I D I プレーヤー）で処理されるが、この点については公知技術であるため詳しく説明しない。なお、A E M 演奏データに基づく奏法波形生成プロセスの処理には相応の遅れ時間があるのに対して、M I D I プレーヤーによる通常の M I D I 演奏データの処理には実質的な時間遅れは生じない。そこで、M I D I プレーヤーによる通常の M I D I 演奏データの処理を、A E M 演奏データに基づく奏法波形生成プロセスの処理における遅れ時間に合わせて、適宜遅延させることにより、両者の再生音の正確な同期をとるようにしている。

【0035】

・省略されている必要情報を補充する機能：

更に、イージー・プレーヤー部 20 で行う機能の一つとして、読み込んだ自動演奏データ（曲ファイル）における A E M 演奏イベントデータ（奏法イベント）の中に、必要な奏法パラメータが含まれていなかった場合、これを補充する機能がある。例えば、或る奏法イベントの奏法 I D の種類が「ビブラート・ロング・ボディ」の場合、ビブラート深さなどの制御パラメータが設定されないと、どの程度のビブラートで奏法波形を合成してよいのか判らない、という不都合がある。そのため、奏法モジュールの種類によっては、奏法イベントには奏法 I D のみならず、必要なパラメータも付属されていなければならないものがある。そこで、イージー・プレーヤー部 20 では、読み込んだ自動演奏データの中の各奏法イベントにおいてそのような必要なパラメータが含まれているかどうかをチェックし、含まれていない場合は、必要なパラメータを自動的に補充する。この情報補充機能は図 6 におけるブロック 302 のタイミングで実行される。イージー・プレーヤー部 20 から奏法シーケンス部 21 に送付される前記奏法イベント列オブジェクトにおいては、そのように補充されたパラメータを含む。必要なパラメータの自動補充の仕方としては、奏法モジュールの各種類毎に、各種のパラメータ毎の所定のデフォルト値（標準値）を用意しておき、このデフォルト値を使用すればよい。デフォルト値の決め方としては、それぞれ所定の固定値を予め用意しておいてもよいし、あるいは、当該パラメータに関する前回使用した値（直前使用値）をバッファしておき、バッファされた直前使用値をデフォルト値として使用するようにしてもよく、その他、適宜の手段を採用してよい。

【0036】

イージー・プレーヤー部 20 で費やされる前記ブロック 301, 302 の処理時間として所定時間が予め確保されており、「P L A Y」指示がなされたときから該所定時間内でこれらのブロック 301, 302 の処理が行われる。ユーザは、「P L A Y」指示を行う前に、奏法波形生成プロセスにおける各種の処理時間情報を設定したり、その他の種々の設定操作を行うことができるが、この点についての説明は省略する。イージー・プレーヤー部 20 は、時間ブロック 302 の最後で、奏法シーケンス部 21 に対して、シーケンス再生開始指示を与える。実際の演奏が開始された後は、波形合成部 24 から再生演奏の現在時間（現在の再生位置）を示す情報をイージー・プレーヤー部 20 で受け取り、この再生演奏の現在時間を表示する。図 6 におけるブロック 303a, 303b, 303c, ... は

10

20

30

40

50

、定期的な割込み処理として実行される表示変更処理（再生演奏の現在時間を表示する処理）のタイミングを示している。

【 0 0 3 7 】

（ 2 ）奏法シーケンス部 2 1

奏法シーケンス部 2 1 は、イージー・プレーヤー部 2 0 から与えられるタイムスタンプ付きの奏法イベント列オブジェクト（つまり奏法イベントデータ列）をバッファし、これをタイムスタンプの時刻つまり再生時刻に従って、順次に読み出す。この順次読出しは、任意に設定される「送り出し頻度」に相当する時間 毎にバッチ処理で行われる。図 6 において、時間 の間隔が例示されている。

・パート管理：

奏法波形の合成は演奏パート別に並列的に行われるので、奏法シーケンス部 2 1 の最初の仕事は、今回の曲演奏において A E M 演奏パートがどれだけあるかを事前に（シーケンス再生開始指示の受理前に）解釈し、必要な数の A E M 演奏の再生パート数を設定するようパート部 2 2 を開設する指示を行うことである。そして、イージー・プレーヤー部 2 0 から与えられるタイムスタンプ付きの奏法イベント列オブジェクト（つまり奏法イベントデータ列）を解釈して、これをパート別にセット（バッファ）しておく。よって、シーケンス再生開始指示の受理後の、奏法イベント列オブジェクト（つまり奏法イベントデータ列）の順次読出しは、各パート毎に行われる。よって、以下の説明で、「或るイベントの次のイベントを先読みする」というような場合、同じパートにおける奏法イベントデータ列を対象してイベントの前後を論じているのである。以下、単に「パート」という言うが、これは「A E M 演奏パート」のことである。

【 0 0 3 8 】

・他の事前処理：

上記パート管理以外の事前処理として、奏法シーケンス部 2 1 においては、各種の動作時間遅れを考慮した時間パラメータに応じた処理を行う。この種の時間パラメータには次のようなものがある。

「奏法シーケンス部での送り出し頻度の時間 」...これは、前述の通り、奏法シーケンス部 2 1 がどの程度の頻度で演奏イベントデータをパート部 2 2 以降の処理に送り出すかを設定するデータである。すなわち、或る送り出し時点（現在時刻）において、この時間の幅内に存在する演奏イベントデータを一括して次段に送り出す。なお、この実施例では、奏法シーケンス部 2 1 は現在時刻の演奏イベント（詳しくは現在処理すべき時間の幅内に存在するイベント）のみならず、それよりも先の（未来の）1 又は複数の演奏イベントをも先行して読み出して、パート部 2 2 に与えるようになっている。これについては追って詳しく説明する。パート部 2 2 以降の処理では、この時間 毎に与えられる演奏イベントデータに基づき奏法波形再生に必要な処理を行う。

【 0 0 3 9 】

「奏法シーケンス部での先送り時間 」...これは、奏法シーケンス部 2 1 がどの程度先の情報まで先行して処理するかを設定する時間である。

「発音潜伏 (l a t e n c y) 時間 」...これは、再生開始時の動作遅延時間に見合った時間である。再生開始シーケンス位置よりもこの発音潜伏 (l a t e n c y) 時間 だけ手前に、波形合成部 2 4 における動作開始時刻をみかけ上設定する（つまり再生開始シーケンス位置を遅らせる）。再生開始時においては、上記送り出し頻度の時間 の幅分の演奏イベントのみならず上記先送り時間 分の演奏イベントについても奏法波形合成処理を一挙に行う必要があるため、その負荷を考慮して、それに相当する遅れ時間を発音潜伏 (l a t e n c y) 時間 として、再生開始時にオフセット設定することで、再生開始時の動作遅れを調整する。

【 0 0 4 0 】

「コードブックデータアクセスでのプリフェッチ時間 」...これは、波形合成部 2 4 が、現在時刻よりもどの程度先のデータまでをハードディスク（その中のコードブック）から R A M 内に読み込むか（プリフェッチするか）を設定する時間である。奏法シーケンス部

10

20

30

40

50

21では、このプリフェッチ時間 のデータを波形合成部24にセットする。

「オーディオデバイスへの出力潜伏 (latency) 時間 」...これは、波形合成部24が、出力用オーディオデバイスの時刻よりもどれだけ手前の時刻から合成処理を行うかを設定する時間である。奏法シーケンス部21では、この出力潜伏 (latency) 時間 のデータを波形合成部24にセットする。例えば、波形合成部24では、合成した波形データを、この出力潜伏 (latency) 時間分だけ先の時刻で使用される出力バッファに書き込むように制御する。

上記各時間パラメータは、それぞれ所定値に固定されていてもよいし、あるいは、ユーザによって可変設定できてよい。ユーザが時間パラメータを設定する場合は、イージー・プレーヤー部20での設定処理によってこれを行う。

10

【0041】

・未来イベントの先読み処理：

奏法シーケンス部21で取り扱うメモリ内にバッファされた或る1パートの奏法イベント列オブジェクトが図7の(a)のようであるとする。EV1, EV2, ...が各イベントを示し、Ts1, Ts2, ...がそれぞれに対応するタイムスタンプを示す。このメモリの読出しポイントの初期値は、初期時点t0に対応しているものとする。

図7の(b)は、奏法シーケンス部21によるイベント処理タイミングを略示するタイミングチャートである。最初のイベント処理タイミングは、イージー・プレーヤー部20からシーケンス再生開始指示が与えられたとき到来する。これを初期時点t0とする。以後のイベント処理タイミングは、送り出し頻度の時間 が経過する毎に到来する。これらを時点t1, t2, ...とする。初期時点t0での処理において、基本的に、時間 分の時間帯(現在時間帯)に存在するイベント(現在イベント)をそのタイムスタンプと共に図7(a)のメモリから読み出す。もちろん、前述の通り、1つのイベントには奏法IDと奏法パラメータ等が含まれているので、当該イベントについてのこれらのデータ全てがワンセットで読み出される。図7(b)の例では、最初のイベント(EV1, EV2)が最初の時間 分の時間帯に存在している例を示している。なお、最初のイベントは通常、アタック部の奏法イベント(EV1)であり、これは前述の通りノートオンイベント(EV2)と対になって一緒に処理される。もちろん、現在時間帯にイベントが存在しない場合もありうる。なお、各イベントEV1, EV2, ...が存在する時間位置は、それぞれのタイムスタンプTs1, Ts2, ...から判明する。

20

30

【0042】

この実施例では、そのような現在の奏法イベントの読出しに加えて、その次に存在する1又は複数の奏法イベント(未来イベント)も読み出す。図7(b)の例では、次の奏法イベントEV3が現在の奏法イベント(EV1, EV2)の読出し時に一緒に読み出される。こうして、読み出された現在及び未来の奏法イベントEV1, EV2, EV3が、それぞれタイムスタンプを伴って、次のパート部22に渡される。この処理は各パート毎にそれぞれ行われる。これらの処理が終わると奏法シーケンス部21は待機状態となる。

【0043】

このように現在の奏法イベントのみならず、その先の(未来の)奏法イベントをも取得して、パート部22に与えるようにしたことにより、パート部22以降の処理において現在の奏法イベントに対応する奏法モジュールの波形合成を行うときに、時間的に相前後する演奏イベントに基づく奏法モジュールの相互関係を考慮に入れて波形合成を行うことができる。たとえば、相前後する演奏イベントに基づく奏法モジュールに基づく奏法波形同士が円滑に接続されるように適切な加工を施すことができるようになる。

40

【0044】

図7の例において、次の処理時点t1が到来すると、奏法シーケンス部21の前記処理が再開される。イベントが存在しない場合は何も処理されない。処理時点tnが到来して奏法シーケンス部21の前記処理が再開されたとき、該処理時点tnにおける現在時間帯に奏法イベントEV3が存在する場合、時間 分の時間帯(現在時間帯)に存在するイベント(現在イベント)を読み出すと共にその次に存在するイベント(未来イベント)を先読

50

みする処理を行う。なお、パート部 2 2 において、受け取ったイベントを記憶しておき、そのイベント処理が行なわれるまで保存しておくようにするので、既に読み出し済みのイベントはもう読み出さなくてよい。例えば、図 7 (b) の例では、処理時点 t_n における現在時間帯にはイベント E V 3 が存在するが、これは前回読み出し時において未来イベントとして読み出されているので、もう読み出さなくてもよい。一方、更に次のイベント E V 4 がそれに先行するイベント E V 3 (現在イベント) の処理にとって必要であれば、これはタイムスタンプと共に読み出してパート部 2 2 に供給する。この場合、パート部 2 2 では、既に受け取り済みの現在イベント E V 3 と今回受け取った未来イベント E V 4 とから、それぞれに対応する奏法モジュールの相互接続関係を調整する。

図 6 において、ブロック 3 1 1 a , 3 1 1 b , 3 1 1 c ... が、奏法シーケンス部 2 1 による上述のイベント読み出し処理 (現在イベントと未来イベントの読み出しを含む) が実行されるタイミングを例示している。

【 0 0 4 5 】

(3) パート部 2 2

パート部 2 2 は、奏法シーケンス部 2 1 から送付された奏法イベント (タイムスタンプ付き) を記憶し、これに基づき所定の「リハーサル」処理を行うと共に、次段の奏法合成部 2 3 における処理を管理する。これらの処理も各パート毎にそれぞれ行われる。図 6 において、ブロック 3 2 1 a , 3 2 1 b , 3 2 1 c ... が、パート部 2 2 において「リハーサル」処理等が実行されるタイミングを例示している。

・リハーサル処理：

「リハーサル」処理とは、奏法合成後の時間的に相前後する各波形構成要素 (T i m b r e , A m p l i t u d e , P i t c h 等) 同士の始点や終点についての時刻やレベル値が滑らかにつながるように、実際の奏法合成を行なう前に、奏法イベントに応じたベクトル I D や代表点値例やその他パラメータなどをリハーサルで読み出し、これに基づき模擬的な奏法合成を行い、各奏法モジュールの始点や終点についての時刻やレベル値を制御するパラメータを適切に設定する処理である。この「リハーサル」処理に基づいて設定されたパラメータを用いて、奏法合成部 2 3 で奏法合成処理を行うことにより、時間的に相前後する奏法波形が各構成要素 (T i m b r e , A m p l i t u d e , P i t c h 等) 毎に滑らかに接続されることになる。すなわち、合成済みの奏法波形若しくは波形構成要素を調整・制御してそれらの奏法波形若しくは波形構成要素同士が滑らかに接続されるようにするのではなく、個々の奏法波形若しくは波形構成要素を合成する直前における「リハーサル」処理によって模擬的に奏法波形若しくは波形構成要素を合成する処理を行ない、始点や終点等の時刻やレベルなどについての最適なパラメータを設定し、これらの最適なパラメータを用いて奏法波形若しくは波形構成要素の合成を行なうことにより結果的に奏法波形若しくは波形構成要素同士が滑らかに接続されるようにするのである。

【 0 0 4 6 】

リハーサル処理では、処理対象である奏法モジュールの種類若しくは性格に応じて必要な処理を行なう。なお、リハーサル処理の対象となるのは主に奏法イベントであるが、ノートオンイベントやノートオフイベントの発生時刻も考慮される。参考のために、発音開始から終了に至るまでの奏法モジュールの組み合わせ状態の一例を図 8 に示す。図 8 におけるイベントの並び順は、(1) アタック (エントランス) モジュールを指定する奏法イベント E V 1、(2) ノートオンイベント E V 2、(3) ボディモジュールを指定する奏法イベント E V 3、(4) ジョイントモジュールを指定する奏法イベント E V 4、(5) ノートオンイベント E V 5、(6) ボディモジュールを指定する奏法イベント E V 6、(7) リリース (フィニッシュ) モジュールを指定する奏法イベント E V 7、(8) ノートオフイベント E V 8、である。

【 0 0 4 7 】

図 8 の例において、現在イベントである奏法イベント E V 1 及びノートオンイベント E V 2 によって指定されるアタック部 (エントランス部) の奏法モジュールに従う奏法波形を合成するとき、前述のようにその次の奏法イベント E V 3 が未来イベントとして先読み

10

20

30

40

50

して与えられ、リハーサル処理では、両者が滑らかに接続されるように必要なパラメータを決定する。次に、その後、奏法イベントE V 3を現在イベントとしてそれによって指定されるボディ部の奏法モジュールに従う奏法波形を合成するときに、その次のジョイント部の奏法イベントE V 4及びそれと対を成すノートオンイベントE V 5が未来イベントとして先読みして与えられ、リハーサル処理では、両者が滑らかに接続されるように必要なパラメータを決定する。同様に、奏法イベントE V 4及びノートオンイベントE V 5を現在イベントとしてそれによって指定されるジョイント部の奏法モジュールに従う奏法波形を合成するときに、前述のようにその次の奏法イベントE V 6が未来イベントとして先読みして与えられ、リハーサル処理では、両者が滑らかに接続されるように必要なパラメータを決定する。なお、この例では、最初のノートオンイベントE V 2で発音される音と次のノートオンイベントE V 5で発音される音とをジョイント部の奏法波形を用いて接続している。同様に、更にその次の奏法イベントE V 6を現在イベントとしてそれによって指定されるボディ部の奏法モジュールに従う奏法波形を合成するときに、その次のリリース部の奏法イベントE V 7及びそれと対を成すノートオフイベントE V 8が未来イベントとして先読みして与えられ、リハーサル処理では、両者が滑らかに接続されるように必要なパラメータを決定する。更に奏法イベントE V 7及びノートオフイベントE V 8を現在イベントとしてそれによって指定されるリリース部（フィニッシュ部）の奏法モジュールに従う奏法波形を合成するときには、消音でその奏法波形が終るが故に、次の波形との接続を考慮する必要がないので、次の奏法イベントを考慮したりリハーサル処理は行なわれない。

10

20

【0048】

次に、いくつかの種類のアタックモジュールに関する「リハーサル」処理の具体例につき説明する。

アタック（エントランス）部モジュール

図9（a）は、現在イベントがアタック（エントランス）部の奏法モジュールの場合のリハーサル処理の一手順例を示すフロー図である。

ステップS 1 aでは、今回処理すべき奏法イベント（現在イベント）（図8の例ではE V 1, E V 2）を奏法合成部23に渡し、その奏法ID（特定のアタック奏法モジュールを指示する）に応じたベクトルID及び代表点値列（Shape）及びその他パラメータを、奏法合成部23の処理によって、前記奏法テーブルからリハーサルとして読み出して、これらをパート部22が受け取り、これに基づき次のようにレベルや時刻等のパラメータ（制御データ）を決定若しくは調整する処理を行なう。

30

【0049】

ステップS 2 aでは、次の奏法イベント（先読みによって得た未来イベント）（図8の例ではE V 3）を奏法合成部23に渡し、その奏法ID（特定のボディ部の奏法モジュールを指示する）に応じたベクトルID及び代表点値列（Shape）及びその他パラメータを、奏法合成部23の処理によって、前記奏法テーブルからリハーサルとして読み出して、これらをパート部22が受け取り、これに基づき次のようにレベルや時刻等のパラメータ（制御データ）を決定若しくは調整する処理を行なう。

ステップS 3 aでは、取得した今回及び次回の奏法イベントについての各データに基づき、今回の奏法イベントに係る奏法モジュールについての所定のデータ（レベルと時刻）を決定する処理を行なう。前ステップS 1 a, S 2 aでは、ここでの処理のために必要なデータを奏法テーブルから読み出せばよい。ここでのリハーサル処理の内容例を説明するために、図10（a）,（b）,（c）,（d）を参照する。

40

【0050】

図10（a）は、アタック奏法モジュールにおける調和成分用の各ベクトル例を示し、「HA」は調和成分振幅ベクトルの代表点値列（一例として0, 1, 2の3点からなる）、「HP」は調和成分ピッチベクトルの代表点値列（一例として0, 1, 2の3点からなる）、「HT」は調和成分波形ベクトルの一例（ただし波形をそのエンベロープのみで略示している）を示す。ちなみに、調和成分波形ベクトルHTは、基本的には音の立ち上がり

50

部の全波形のサンプルデータからなり、その最後の部分にループ波形を含んでいる。このループ波形は後続波形と接続するときのクロスフェード合成のためにループ読み出しされるものである。アタック奏法モジュールにおける調和成分の開始時刻を定義するパラメータ“preBlockTimeE”は、実際の発音開始時点と、アタック波形の調和成分の波形発生開始時点とのずれを規定するパラメータである。開始時刻の設定は、これと対を成すノートオンイベント（図8の例ではEV2）のタイムスタンプを取得して、実際の発音開始時刻（図10における“noteOnTime”）を把握し、これと“preBlockTimeE”との差“noteOnTime - preBlockTimeE”を調和成分のアタック奏法モジュール開始時刻“startTimeH”として設定する。

【0051】

アタック奏法モジュールにおける調和成分の終了時刻を定義するパラメータのうち、“postBlockTimeE”は、実際の発音開始時点とアタック波形の調和成分の本体波形終了時点とのずれを規定するパラメータであり、“fadeTimeE”は、アタック波形の末尾のクロスフェード時間幅を規定するパラメータである。従って、末尾のクロスフェード部分を含めたアタック奏法モジュールにおける調和成分の終了時刻“endTimeH”は、“noteOnTime + (postBlockTimeE + fadeTimeE)”として求まる。この終了時刻“endTimeH”が、次の奏法イベント（図8の例ではEV3）の調和成分のモジュール開始時刻を規定するデータとして奏法合成部23に返される。こうして、調和成分の終了時刻“endTimeH”に応じて次の奏法モジュールの調和成分の開始時刻を設定するリハーサル処理が行なわれる。

【0052】

図10(b)は、アタック奏法モジュールにおける調和外成分用の各ベクトル例を示し、「NHA」は調和外成分振幅ベクトルの代表点値列（一例として0, 1の2点からなる）、「NHT」は調和外成分波形ベクトルの一例（ただし波形をそのエンベロープのみで略示している）を示す。アタック奏法モジュールにおける調和外成分の開始時刻を定義するパラメータ“preTimeNH”は、実際の発音開始時点と、アタック波形の調和外成分の波形発生開始時点とのずれを規定するパラメータである。調和外成分の開始時刻の設定も、前述の調和成分の場合と同様に、それと対を成すノートオンイベント（図8の例ではEV2）のタイムスタンプを取得して、実際の発音開始時刻（図10における“noteOnTime”）を把握し、これと“preTimeNH”との差“noteOnTime - preTimeNH”を調和外成分のアタック奏法モジュール開始時刻“startTimeNH”として設定する。

【0053】

アタック奏法モジュールにおける調和外成分の終了時刻を定義するパラメータ“postTimeNH”は、実際の発音開始時点とアタック波形の調和外成分の終了時点とのずれを規定するパラメータである。アタック奏法モジュールにおける調和外成分の終了時刻“endTimeNH”は、“noteOnTime + postTimeNH”として求まる。この終了時刻“endTimeNH”が、次の奏法イベント（図8の例ではEV3）の調和外成分のモジュール開始時刻を規定するデータとして奏法合成部23に返される。こうして、調和外成分の終了時刻“endTimeNH”に応じて次の奏法モジュールの調和外成分の開始時刻を設定するリハーサル処理が行なわれる。このように、調和外成分の時刻合わせは、調和成分とは独立に行なわれる。

【0054】

レベルについてのリハーサル処理は、現在の奏法イベントに係るアタック部奏法モジュールの振幅ベクトルの終点（図10(a)のHAの2位置）のレベル（振幅値）とピッチベクトルの終点（図10(a)のHPの2位置）のレベル（ピッチの値）を、次の奏法イベントに係るボディ部奏法モジュールの振幅ベクトルの始点（図10(c)のHAの0位置）のレベル（振幅値）とピッチベクトルの始点（図10(c)のHPの0位置）のレベル（ピッチの値）に合わせることを行なう。

図10(c)は、ボディ部の奏法モジュールにおける調和成分用の各ベクトル例を示し、

10

20

30

40

50

「H A」は調和成分振幅ベクトルの代表点値列（一例として0, 1の2点からなる）、「H P」は調和成分ピッチベクトルの代表点値列（一例として0, 1の2点からなる）、「H T」は調和成分波形ベクトルの一例（ただし波形は黒塗した矩形で略示している）を示す。なお、ボディ部の調和成分波形ベクトルH Tは、図で黒塗したN個の矩形部分（0, 1, …… N - 1）にそれぞれ対応するN個のループ波形からなり、これらのループ波形を順次にループ読み出しながら順次切り替えて接続していくことで所定の時間長のボディ部波形が生成される。ボディ部の時間長を少しだけ短縮又は伸張したい場合は、各ループ部のループ時間を短縮/伸張すればよい。また、ボディ部の時間長をそれ以上に短縮したい場合は、該N個のループ波形のうちの任意の1乃至複数を適宜間引いて読み出せばよい。一方、ボディ部の時間長をそれ以上に伸張したい場合は、該N個のループ波形のうちの任意の複数を所定の順序であるいはランダムに該N個のループ波形の間に挿入すればよい。

10

【0055】

図9(a)のステップS2aにおいては、調和成分振幅ベクトルについては、次の奏法イベントに係るボディ部の奏法モジュールの調和成分振幅ベクトルH Aの始点（図10(c)のH Aの0位置）のレベルデータを奏法テーブルから取得する。そして、次のステップS3aにおいて、奏法テーブルから取得したレベルデータにベロシティ値やボリューム設定値などを加味して、ボディ部の奏法モジュールの調和成分振幅ベクトルH Aの始点の実際のレベルを算出し、これを、現在の奏法イベントに係るアタック部奏法モジュールの調和成分振幅ベクトルの終点（図10(a)のH Aの2位置）のレベル（振幅値）として設定する。

20

同様に、調和成分ピッチベクトルについても、次の奏法イベントに係るボディ部の奏法モジュールの調和成分ピッチベクトルH Pの始点（図10(c)のH Pの0位置）のピッチ値データを奏法テーブルから取得する。そして、奏法テーブルから取得したピッチ値データにピッチ制御値を加味して、ボディ部の奏法モジュールの調和成分ピッチベクトルH Pの始点の実際のピッチを算出し、これを、現在の奏法イベントに係るアタック部奏法モジュールの調和成分ピッチベクトルの終点（図10(a)のH Pの2位置）のレベル（ピッチ値）として設定する。

【0056】

図10(d)は、ボディ部の奏法モジュールにおける調和外成分用の各ベクトル例を示し、「N H A」は調和外成分振幅ベクトルの代表点値列（一例として0, 1の2点からなる）、「N H T」は調和外成分波形ベクトルの一例を略記して示す。調和外成分波形ベクトルN H Tは3つのブロックからなり、第1のブロックは所定の期間“N H B l o c k T i m e 0”分の調和外成分波形の全波形からなり、第2のブロックはループ用の調和外成分波形（L o o p）からなり、第3のブロックは所定の期間“N H B l o c k T i m e 1”分の調和外成分波形の全波形からなる。ボディ部の自感知用に応じた調和外成分波形の時間長の調節は、第2のブロックのループ用の調和外成分波形（L o o p）をループ再生する時間長を調節することで行なう。

30

図9(a)のステップS2aにおいては、調和外成分振幅ベクトルについても、次の奏法イベントに係るボディ部の奏法モジュールの調和外成分振幅ベクトルN H Aの始点（図10(d)のN H Aの0位置）のレベルデータを奏法テーブルから取得する。そして、次のステップS3aにおいて、奏法テーブルから取得したレベルデータにベロシティ値やボリューム設定値などを加味して、ボディ部の奏法モジュールの調和外成分振幅ベクトルN H Aの始点の実際のレベルを算出し、これを、現在の奏法イベントに係るアタック部奏法モジュールの調和外成分振幅ベクトルの終点（図10(b)のN H Aの1位置）のレベル（振幅値）として設定する。

40

【0057】

上記の処理が終了すると、図9(a)のステップS4aにおいて、現在の奏法イベントに係るアタック部奏法モジュールの合成を開始する指示を、奏法合成部23に対して与える。次のステップS a 5では、上記のようにして決定された今回の奏法イベント（図8の例ではE V 1）の調和成分及び調和外成分の終了時刻“e n d T i m e H”、“e n d T i

50

m e N H ” を、次の奏法イベント（図 8 の例では E V 3 ）の調和成分及び調和外成分のモジュール開始時刻を規定するデータとして設定する。リハーサル処理が終了すると、図 5 のフローは奏法合成部 2 3 の処理に移行する。奏法合成部 2 3 の処理については追って説明することとし、次に、更に別のモジュールについてのリハーサル処理について説明する。

【 0 0 5 8 】

ボディ部モジュール

図 9（b）は、現在イベントがボディ部の奏法モジュールの場合のリハーサル処理の一手順例を示すフロー図である。

ステップ S 1 b では、今回処理すべき奏法イベント（現在イベント）（例えば図 8 の E V 3 又は E V 6 ）を奏法合成部 2 3 に渡し、その奏法 I D（特定のボディ部の奏法モジュールを指示する）に応じたベクトル I D 及び代表点値列（ S h a p e ）及びその他パラメータを、奏法合成部 2 3 の処理によって、前記奏法テーブルからリハーサルとして読み出して、これらをパート部 2 2 が受け取り、これに基づき次のようにレベルや時刻等のパラメータ（制御データ）を決定若しくは調整する処理を行なう。なお、前回の奏法イベントのリハーサル処理のときに既に調整・変更されているパラメータは、調整・変更されているものを用いる。

【 0 0 5 9 】

ステップ S 2 b では、次の奏法イベント（先読みによって得た未来イベント）を奏法合成部 2 3 に渡し、その奏法 I D に応じたベクトル I D 及び代表点値列（ S h a p e ）及びその他パラメータを、奏法合成部 2 3 の処理によって、前記奏法テーブルから引き出して、これらをパート部 2 2 が受け取り、これに基づき次のようにレベルや時刻等のパラメータ（制御データ）を決定若しくは調整する処理を行なう。ボディ部の次の来る奏法イベントは、リリース部の奏法イベントがジョイント部の奏法イベントである。図 8 では、ボディ部奏法イベント E V 3 の次にジョイント部奏法イベント E V 4 及びそれと対を成すノートオンイベント E V 5 が来ており、ボディ部奏法イベント E V 6 の次にリリース部奏法イベント E V 7 及びそれと対を成すノートオフイベント E V 8 が来ている。

ステップ S 3 b、S 5 b では、取得した今回及び次回の奏法イベントについての各データに基づき、今回の奏法イベントに係る奏法モジュールについての所定のデータ（時刻とレベル）を決定若しくは調整する処理を行なう。ステップ S 4 b においては、今回の奏法イベントに係るボディ部奏法モジュールの合成を開始する指示を、奏法合成部 2 3 に対して与える。

【 0 0 6 0 】

ボディ部の奏法モジュールについては、基本的に、その開始時刻はそれに先行する奏法モジュールの終了時刻に合わせ、その終了時刻はそれに後続する奏法モジュールの開始時刻に合わせるようにし、また、調和成分及び調和外成分の各振幅ベクトル及び調和成分のピッチベクトルにおける各始点のレベル及び終点のレベルは、ボディ部奏法モジュールのものを使用する。つまり、先行する奏法モジュールの終点のレベルをボディ部奏法モジュールの始点のレベルに合わせ、後続する奏法モジュールの始点のレベルをボディ部奏法モジュールの終点のレベルに合わせるように、リハーサル処理を行なう。

ボディ部の奏法モジュールの調和成分及び調和外成分の各開始時刻は、その前の奏法イベント（例えば前述したアタック部奏法モジュールのイベント）でのリハーサル処理によって既に決定されているので、ここでは求める必要がない。

ボディ部の奏法モジュールの調和成分及び調和外成分の各終了時刻を決定するために、次の奏法モジュール（リリース部又はジョイント部）をステップ S 2 b でリハーサルして、その調和成分及び調和外成分の各開始時刻を求める。そして、ステップ S 3 b では、上記求めた次の奏法モジュール（リリース部又はジョイント部）の調和成分及び調和外成分の各開始時刻を、今回のボディ部の奏法モジュールの調和成分及び調和外成分の各終了時刻として決定する。

【 0 0 6 1 】

10

20

30

40

50

このような時刻決定処理の詳細は、アタック部奏法モジュールに関連して既に説明したやり方に準じたやり方で行なえばよい。

参考のために、次の奏法モジュールとなりうるジョイント部奏法モジュールにおける調和成分用の各ベクトル例を図11(a)に示し、同ジョイント部奏法モジュールにおける調和外成分用の各ベクトル例を図11(b)に示す。HA, HP, HT, NHA, NHT等の符号の意味は、図10におけるものと同様である。図11(a)において、調和成分の開始時刻を定義するパラメータ“preTimeH”は、ジョイント部でのノートオンイベント発生時点(図8の例ではEV5)と、ジョイント部の調和成分の波形発生開始時点とのずれを規定するものである。このリハーサルによって、更にその次のイベントであるノートオンイベント(図8の例ではEV5)のタイムスタンプを取得して、実際の発音開始時刻(図11(a)における“noteOnTime”)を把握し、これと“preTimeH”との差“noteOnTime - preTimeH”を調和成分のジョイント部奏法モジュール開始時刻として設定する。こうして、リハーサルで設定された調和成分のジョイント部奏法モジュール開始時刻を、ボディ部の奏法モジュールの調和成分の終了時刻として決定する。図11(b)における調和外成分の開始時刻を定義するパラメータ“preTimeNH”についても同様にリハーサル処理することで、調和外成分のジョイント部奏法モジュール開始時刻を設定し、ボディ部の奏法モジュールの調和外成分の終了時刻として決定する。

【0062】

また、ボディ部の奏法モジュールの調和成分及び調和外成分の各振幅ベクトル及び調和成分のピッチベクトルにおける各始点のレベルを、その前の奏法イベントの奏法モジュール(例えば前述したアタック部奏法モジュール)の調和成分及び調和外成分の各振幅ベクトル及び調和成分のピッチベクトルにおける各終点のレベルとしてそれぞれ設定することは、前の奏法イベントでのリハーサル処理によって既に済んでいる。

よって、ここでは、ボディ部の奏法モジュールの調和成分及び調和外成分の各振幅ベクトル及び調和成分のピッチベクトルにおける各終点のレベルを、次の奏法イベント(未来イベント)に係る奏法モジュールの調和成分及び調和外成分の各振幅ベクトル及び調和成分のピッチベクトルにおける各始点のレベルとして決定する(ステップS5b)。

【0063】

このようなレベル決定処理の詳細も、アタック部奏法モジュールに関連して既に説明したやり方に準じたやり方で行なえばよい。調和成分振幅ベクトルについては、現在の奏法イベントに係るボディ部の奏法モジュールの調和成分振幅ベクトルHAの終点(図10(c)のHAの2位置)のレベルデータを奏法テーブルから取得する。そして、奏法テーブルから取得したレベルデータにベロシティ値やボリューム設定値などを加味して、ボディ部の奏法モジュールの調和成分振幅ベクトルHAの終点の実際のレベルを算出し、これを、次の奏法イベントに係る奏法モジュールの調和成分振幅ベクトルの始点(図11(a)のHAの0位置)のレベル(振幅値)として設定する。また、同様に、調和成分ピッチベクトルについても、現在の奏法イベントに係るボディ部の奏法モジュールの調和成分ピッチベクトルHPの終点(図10(c)のHPの2位置)のピッチ値データを奏法テーブルから取得する。そして、奏法テーブルから取得したピッチ値データにピッチ制御値を加味して、ボディ部の奏法モジュールの調和成分ピッチベクトルHPの終点の実際のピッチを算出し、これを、次の奏法イベントに係る奏法モジュールの調和成分ピッチベクトルの始点(図11(a)のHPの0位置)のレベル(ピッチ値)として設定する。

調和外成分振幅ベクトルについても同様に、現在の奏法イベントに係るボディ部の奏法モジュールの調和外成分振幅ベクトルNHAの終点(図10(d)のNHAの1位置)のレベルデータを奏法テーブルから取得し、取得したレベルデータにベロシティ値やボリューム設定値などを加味して、ボディ部の奏法モジュールの調和外成分振幅ベクトルNHAの終点の実際のレベルを算出し、これを、次の奏法イベントに係る奏法モジュールの調和外成分振幅ベクトルの始点(図11(b)のNHAの0位置)のレベル(振幅値)として設定する。

10

20

30

40

50

【 0 0 6 4 】

ジョイント部モジュール

図9(c)は、現在イベントがジョイント部の奏法モジュールの場合のリハーサル処理の一手順例を示すフロー図である。

ステップS1cでは、今回処理すべき奏法イベント(現在イベント)(例えば図8のEV4, EV5)を奏法合成部23に渡し、その奏法ID(特定のジョイント部の奏法モジュールを指示する)に応じたベクトルID及び代表点値列(Shape)及びその他パラメータを、奏法合成部23の処理によって、前記奏法テーブルからリハーサルとして読み出して、これらをパート部22が受け取り、これに基づき次のようにレベルや時刻等のパラメータ(制御データ)を決定若しくは調整する処理を行なう。なお、前回の奏法イベントのリハーサル処理のときに既に調整・変更されているパラメータは、調整・変更されているものを用いる。

10

【 0 0 6 5 】

ステップS2cでは、次の奏法イベント(先読みによって得た未来イベント)を奏法合成部23に渡し、その奏法IDに応じたベクトルID及び代表点値列(Shape)及びその他パラメータを、奏法合成部23の処理によって、前記奏法テーブルから引き出して、これらをパート部22が受け取り、これに基づき次のようにレベルや時刻等のパラメータ(制御データ)を決定若しくは調整する処理を行なう。ジョイント部の次の来る奏法イベントは、2番目のボディ部の奏法イベント(例えば図8のEV6)である。

ステップS3c、S5cでは、取得した今回及び次回の奏法イベントについての各データに基づき、今回の奏法イベントに係る奏法モジュールについての所定のデータ(時刻とレベル)を決定若しくは調整する処理を行なう。ステップS4cにおいては、今回の奏法イベントに係るジョイント部奏法モジュールの合成を開始する指示を、奏法合成部23に対して与える。

20

【 0 0 6 6 】

ジョイント部の奏法モジュールについては、基本的に、調和成分及び調和外成分の各振幅ベクトル及び調和成分のピッチベクトルにおける各始点のレベルは、先行するボディ部奏法モジュール(例えば図8のEV3)の各終点のレベルに合わせるようにし、調和成分及び調和外成分の各振幅ベクトル及び調和成分のピッチベクトルにおける各終点のレベルは、後続するボディ部奏法モジュール(例えば図8のEV6)の各始点のレベルに合わせるように、リハーサル処理を行なう。

30

ジョイント部の奏法モジュールの調和成分及び調和外成分の各振幅ベクトル及び調和成分のピッチベクトルにおける各始点のレベルのデータは、前の奏法イベントのリハーサル処理によって(図9(b)のステップS5b)既に決定されているので、それを用いる。よって、ステップS3cでは、主に、ジョイント部の奏法モジュールの調和成分及び調和外成分の各振幅ベクトル及び調和成分のピッチベクトルにおける各終点のレベルとして、後続するボディ部奏法モジュール(例えば図8のEV6)の各始点のレベルに見合った値に決定するよう、ステップS2cで取得した次のボディ部奏法モジュールのリハーサル結果を利用して処理を行なう。これは、具体的には、図9(a)のステップS3aにおける処理と同様の手法で行なえばよいため、その詳細説明は省略する。

40

【 0 0 6 7 】

ジョイント部の奏法モジュールの開始時刻については、図11(a)を参照して前述したように、次のイベントであるノートオンイベント(図8の例ではEV5)のタイムスタンプを取得して、実際の発音開始時刻(図11(a)における“noteOnTime”)を把握し、これと“preTimeH”との差“noteOnTime - preTimeH”を調和成分のジョイント部奏法モジュール開始時刻として設定する。調和外成分のジョイント部奏法モジュール開始時刻も、図11(b)を参照して前述した通りの手法で求める。これらは、ステップS1cのリハーサルのときに求めればよい。

また、ジョイント部の奏法モジュールの終了時刻については、図11(a)に示すように、調和成分の終了時刻を定義するパラメータ“postTimeH”がジョイント部での

50

ノートオンイベント発生時点（図8の例ではEV5）と、ジョイント部の調和成分の波形発生終了時点とのずれを規定しているので、その次のイベントであるノートオンイベント（図8の例ではEV5）時刻“noteOnTime”にこれを加算して、“noteOnTime + postTimeH”を調和成分のジョイント部奏法モジュール終了時刻として決定する。ステップS5cでは、こうして、リハーサルで設定された調和成分のジョイント部奏法モジュール終了時刻を、次のボディ部の奏法モジュールの調和成分の開始時刻として設定する。図11（b）における調和外成分の終了時刻を定義するパラメータ“postTimeNH”についても同様にリハーサル処理することで、調和外成分のジョイント部奏法モジュール終了時刻を決定し、次のボディ部の奏法モジュールの調和外成分の開始時刻として設定する。

10

【0068】

リリース（フィニッシュ）部モジュール

図9（d）は、現在イベントがリリース（フィニッシュ）部の奏法モジュールの場合のリハーサル処理の一手順例を示すフロー図である。

ステップS1dでは、今回処理すべき奏法イベント（現在イベント）（例えば図8のEV7, EV8）を奏法合成部23に渡し、その奏法ID（特定のジョイント部の奏法モジュールを指示する）に応じたベクトルID及び代表点値列（Shape）及びその他パラメータを、奏法合成部23の処理によって、前記奏法テーブルからリハーサルとして読み出して、これらをパート部22が受け取り、これに基づき次のようにレベルや時刻等のパラメータ（制御データ）を決定若しくは調整する処理を行なう。なお、前回の奏法イベントのリハーサル処理のときに既に調整・変更されているパラメータは、調整・変更されているものを用いる。通常、前の奏法イベントのときに、既に今回処理すべき奏法イベントについて必要なデータはリハーサル済みであるから、このステップS1dは実際には省略可能である。

20

【0069】

参考のために、リリース部奏法モジュールにおける調和成分用の各ベクトル例を図11（c）に示し、同奏法モジュールにおける調和外成分用の各ベクトル例を図11（d）に示す。HA, HP, HT, NHA, NHT等の符号の意味は、図10と同様である。図11（c）において、調和成分の開始時刻を定義するパラメータのうち、“fadeTimeF”は前の奏法モジュールの末尾波形とリリース部の先頭波形とをクロスフェード合成するための時間を示し、“preBlockTimeF”はクロスフェード終了時から次のイベントであるノートオフイベントの発生時点（図8の例ではEV8）との時間差を示す。リリース部奏法モジュールの調和成分の開始時刻は、ノートオフイベントの発生時刻“noteOffTime”を基準にして、“noteOffTime - (fadeTimeF + preBlockTimeF)”で決定される。また、リリース部奏法モジュールの調和外成分の開始時刻は、“noteOffTime - preTimeNH”で決定される。これらの開始時刻は、その前の奏法イベントであるボディ部奏法モジュールのリハーサル処理（図9（b）のステップS2b, S3b）において既に求められているので、それを用いることができる。

30

また、リリースの奏法モジュールの調和成分及び調和外成分の各振幅ベクトル及び調和成分のピッチベクトルにおける各始点のレベルは、先行するボディ部奏法モジュール（例えば図8のEV6）の各終点のレベルに合わせる。これも、その前の奏法イベントであるボディ部奏法モジュールのリハーサル処理（図9（b）のステップS5b）において既に求められているので、それを用いることができる。

40

【0070】

このように、発音を終了させる奏法モジュールであるリリース（フィニッシュ）部については、それ以前に必要なリハーサルが済んでいる筈であるので、ステップS1dによるリハーサルは、実際には不要である。ステップS4dでは、今回の奏法イベントに係るリリース部奏法モジュールの合成を開始する指示を、奏法合成部23に対して与える。

なお、図9（a）（b）（c）における各ステップS5a, S5b, S5cの処理は、パ

50

ート部 2 2 によるリハーサル処理で行なわずに、次の奏法合成部 2 3 で実際に奏法合成を行なうときに行なうようにしてもよい。

【 0 0 7 1 】

(4) 奏法合成部 2 3

図 5 において、奏法合成部 2 3 は、パート部 2 2 から送付された奏法イベント（タイムスタンプ付き）とリハーサル結果データを受けて所定の奏法合成処理を行なう。この奏法合成処理では、所定の奏法アルゴリズムに従い、奏法イベントの奏法 ID 及びパラメータあるいは制御データを解釈して処理し、これに基づき奏法テーブルから各ベクトル ID と代表点値列及び各種パラメータを読み出し、これらを修飾又は加工もしは変更する。そして、奏法イベントに対応するこれらの各ベクトル ID と代表点値列及び各種パラメータ等と前記リハーサルによって決定された時刻及びレベル等のパラメータ（制御データ）などをパケット化して時間順に従うストリームデータとして出力する。図 6 において、ブロック 3 3 1 a , 3 3 1 b , 3 3 1 c ... が、奏法合成部 2 3 において奏法合成処理等が実行されるタイミングを例示している。また、図 6 のブロック 3 3 0 は上記パケット化されたストリームデータの出力処理部分を例示し、その中のブロック 3 3 0 a , 3 3 0 b , 3 3 0 c ... が各ストリームデータの出力タイミングを例示している。

10

【 0 0 7 2 】

(5) 波形合成部 2 4

図 5 において、波形合成部 2 4 は、各ベクトル ID と代表点値列等の上記パケット化されたストリームデータを奏法合成部 2 3 から受け取り、現在時刻より前記プリフェッチ時間だけ前の時刻に、ベクトル ID に応じて波形データベース内のコードブックから波形テンプレートデータ等を読み出し、また、現在時刻より前記出力潜伏（latency）時間だけ前の時刻に、代表点値列とパラメータ等に基づき振幅ベクトル及びピッチベクトルのエンベロープ波形形状を生成し、これらに基づき奏法波形の調和成分波形と調和外成分波形をそれぞれ形成する。そして、奏法波形の調和成分波形と調和外成分波形をそれぞれの時刻データに応じて所定の時刻位置に貼り付けて、最終的にこれらを加算合成し、奏法波形の合成を行なう。ここで確立される再生時刻（つまり現在時刻）データがイメージ・プレーヤー部 2 0 に与えられ、再生位置（時刻）のリアルタイム表示に用いられる。図 6 において、ブロック 3 4 1 , 3 4 2 , 3 4 3 , ... は、波形合成部 2 4 において波形データベース内のコードブックに対するプリフェッチのタイミングを例示している。なお、波形合成部 2 4 において前記波形テンプレートデータ及び振幅ベクトル及びピッチベクトルのエンベロープ波形形状等に基づき奏法波形データを生成する手法としては、例えば「ソフトウェア音源」として知られた手法を適宜採用することができる。波形合成部 2 4 で合成された奏法波形データは、出力バッファ（図 1 の波形出力部 1 0 8 に含まれる）に与えられる。該出力バッファに記憶された奏法波形データが所定の再生サンプリング周期で読み出されて、サウンドシステム 1 0 8 A（図 1）を經由して空間的に発音される。

20

30

【 0 0 7 3 】

なお、上記実施例では、現在の奏法イベントの読み出しとそれに対応する未来の奏法イベントの先読みを、所定時間 毎に行っているが、これに限らず、随時行うようにしてもよい。

40

【 0 0 7 4 】

【 発明の効果 】

以上の通り、この発明によれば、指定された奏法モジュールに対応する波形を実際に合成する前に、先行する奏法モジュールの少なくとも終端特性に関する終端情報と後続する奏法モジュールの少なくとも先端特性に関する先端情報とを取得する（リハーサルする）ことで、取得した終端情報と先端情報の関係から両者が滑らかに接続されるように、少なくとも一方の特性を修正することができる。このように修正された特性に従い、奏法モジュールに対応する波形を合成することで、相前後する奏法モジュールに基づく奏法波形同士が円滑に接続されるようにすることができる。

【 図面の簡単な説明 】

50

【図 1】この発明の一実施例において利用可能な装置のハードウェア構成例を示すブロック図。

【図 2】奏法モジュールのデータ形式の一例を説明する図。

【図 3】或る奏法モジュールに対応する実波形区間を構成する各成分及び要素の一例を模式的に示す図。

【図 4】1つの曲の自動演奏データ（曲ファイル）の構成例を説明する図。

【図 5】同実施例に係る奏法波形生成プロセスの大まかな手順を示すフローチャート。

【図 6】同実施例に係る奏法波形生成プロセスを構成する各処理ブロックが受け持つ処理の時間関係を大まかに示す概略タイミングチャート。

【図 7】（ a ）は奏法イベント列オブジェクトの一例を示す図、（ b ）は現在の奏法イベントの処理タイミングを規定する送り出し頻度とその後続する奏法イベントを先読みして送り出す先送り時間との関係を例示するタイミングチャート。

10

【図 8】発音開始から終了に至るまでの奏法モジュールの組み合わせ状態の一例を示すタイミングチャート。

【図 9】各種の奏法モジュールに対応するリハーサル処理の例を示すフローチャート。

【図 10】（ a ）及び（ b ）はアタック部の奏法モジュールにおける調和成分及び調和外成分用の各ベクトル例を示す図、（ c ）及び（ d ）はボディ部の奏法モジュールにおける調和成分及び調和外成分用の各ベクトル例を示す図。

【図 11】（ a ）及び（ b ）はジョイント部の奏法モジュールにおける調和成分及び調和外成分用の各ベクトル例を示す図、（ c ）及び（ d ）はリリース部の奏法モジュールにおける調和成分及び調和外成分用の各ベクトル例を示す図。

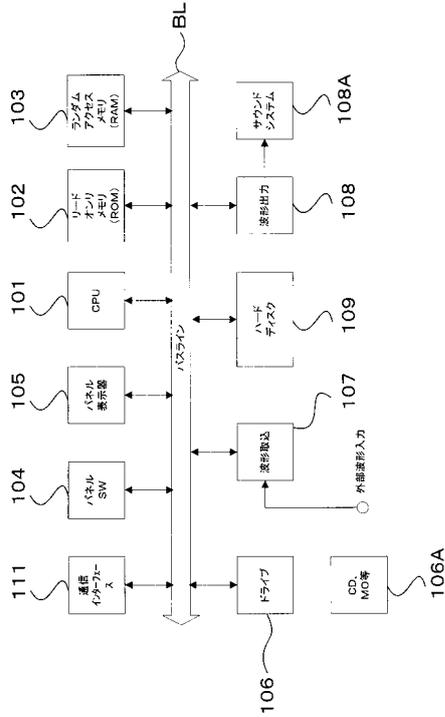
20

【符号の説明】

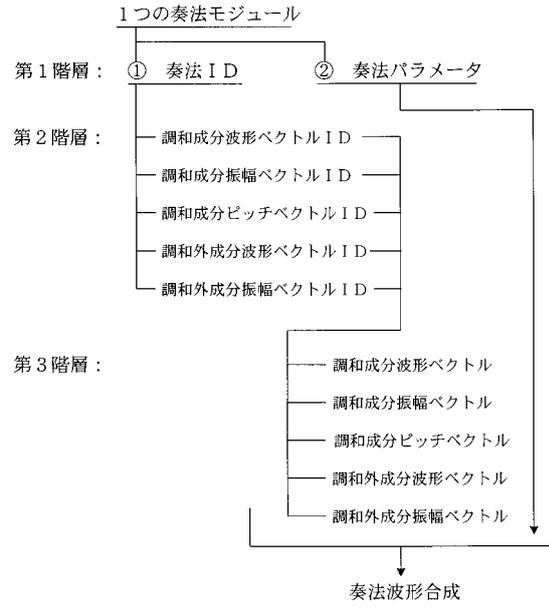
- 1 0 1 CPU
- 1 0 2 ROM
- 1 0 3 RAM
- 1 0 4 パネルスイッチ
- 1 0 5 パネル表示器
- 1 0 8 波形出力部 1 0 8
- 1 0 9 ハードディスク
- 2 0 イージー・プレーヤー部 2 0
- 2 1 奏法シーケンス部
- 2 2 パート部
- 2 3 奏法合成部
- 2 4 波形合成部

30

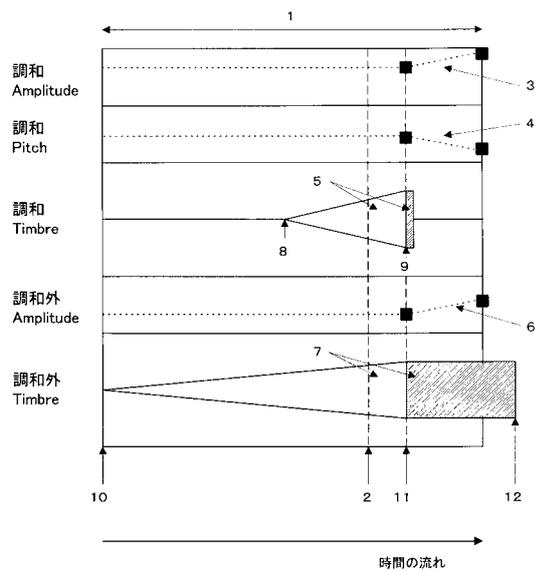
【図1】



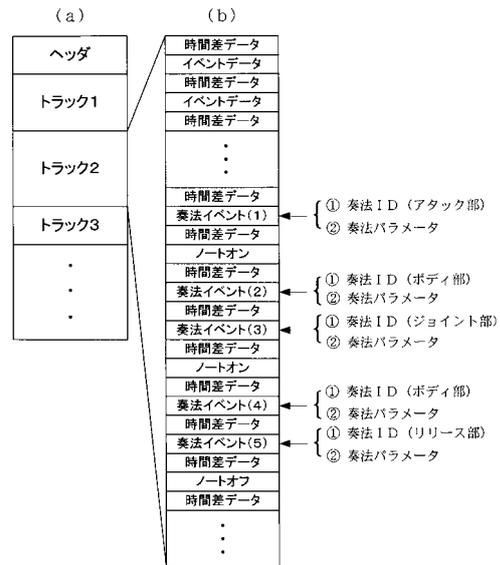
【図2】



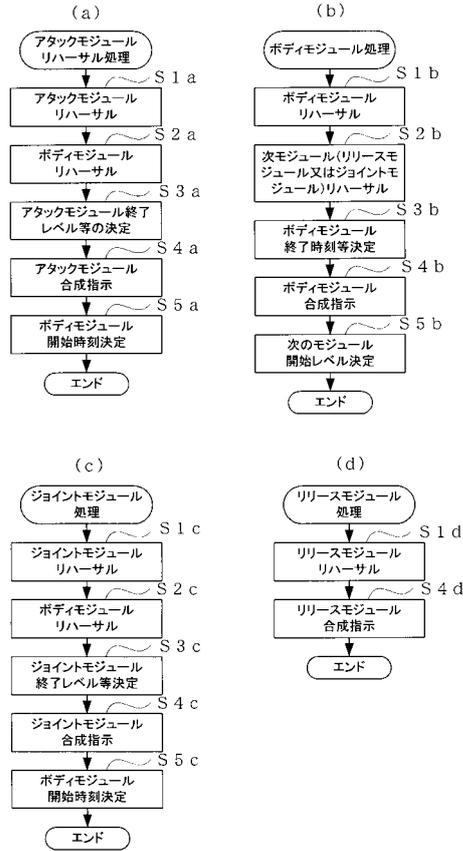
【図3】



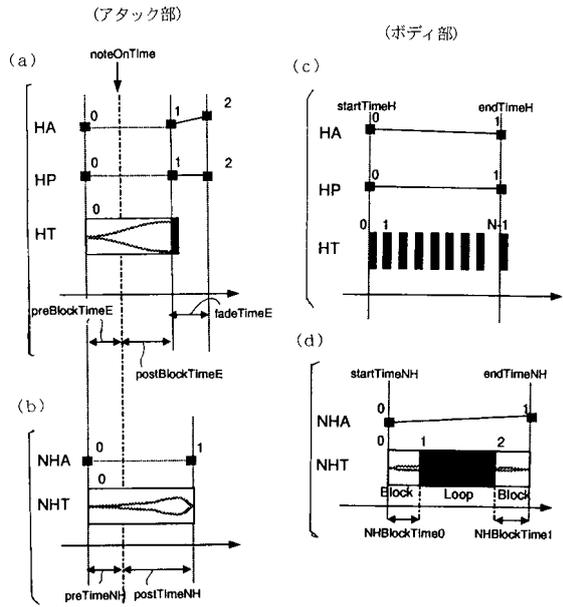
【図4】



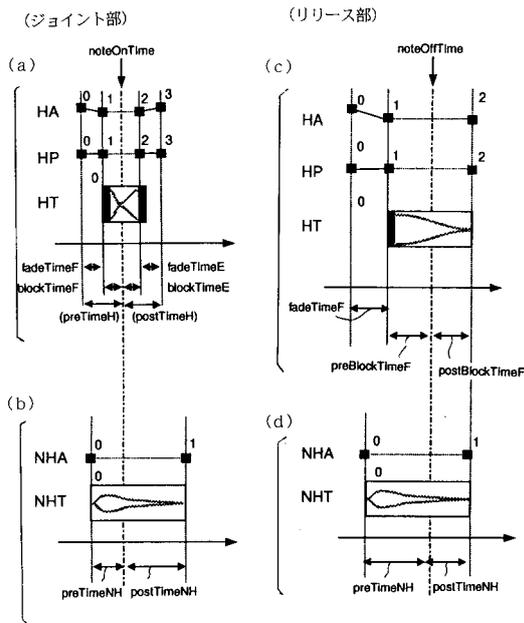
【 図 9 】



【 図 1 0 】



【 図 1 1 】



フロントページの続き

審査官 小宮 慎司

(56)参考文献 特開2000-122666(JP,A)
特開2000-276174(JP,A)

(58)調査した分野(Int.Cl.⁷, DB名)
G10H 1/00 - 7/12