目 次

1 概要

1.1 紹介

v.Connect-STAND[1]は音声合成分析技術 WORLD[1, 2] 及び UTAU[3] 向けに作られた音声データベースを使用して VOCALOID2[4] 用シーケンス[2]から歌声を合成するツールです。コマンドラインを使用した CUI スタイルのアプリケーションですが、バーチャル・ボーカル音源用シーケンサ Cadencii[5] 上から使用することを前提として作られています。

1.2 ライセンス

v.Connect-STAND 及び、内部で使用されている FFTW[6], WORLD0.0.1, libiconv[7] は GNU General Public License(GPLv3) でライセンスされます。またソースコードは GPLv3 の下公開されています。そちらに興味のある方は、SourceForge 内 Cadencii プロジェクト内にソースコードを置いてありますので下記をご参照ください。

http://sourceforge.jp/projects/cadencii/

v.Connect-STAND は Cadencii Project の一部です。内部で使用したライブラリ (FFTW, WORLD, libiconv) を除くソースコードの著作権は HAL 及び kbinani に帰属します。

1.3 免責事項

このツールを使ったことによるいかなる損害も作者は一切の責任を負いません。また、このツールは一切の保証を伴わない「現状渡し」で提供されます。不具合がないことおよび不具合が修正されることを、作者は保証しません。

[1] 2010/11/04 現在、まだリリースしてないよ !!!
[2] 実際には拡張 VSQ メタテキストを使用しています。

2　インストール方法

2.1　ダウンロード

以下の URL から最新版の Cadencii をダウンロードしてください。

Cadencii Wiki http://www9.atwiki.jp/boare/pages/18.html

2.2　インストール

ダウンロードしたファイルを解凍し、Cadencii をインストールしてください。自動的に v.Connect-STAND もインストールされます[3]。

2.3　実行環境

Cadencii の実行、及び v.Connect-STAND の実行には以下の環境が必要です。

- .NET Framework 2.0 以上
- Visual C++ライブラリ DLL

それぞれ以下の URL からダウンロードしてインストールしてください。

- http://msdn.microsoft.com/ja-jp/netframework/aa569263.aspx
- http://support.microsoft.com/default.aspx?scid=kb;EN-US;q259403

なおここに挙げた環境は Windows の場合のみです。
また開発環境は以下の通りです。これよりも低スペックなマシンで実行した場合合成に時間がかかる場合があります。

CPU	Core2Quad 2.8GHz
RAM	4.0GB
HDD	1TB

実行中はメモリをかなり消費します。30 秒の合成に 500MB 程度かかるのでメモリを十分確保した状態での実行をお勧めいたします。

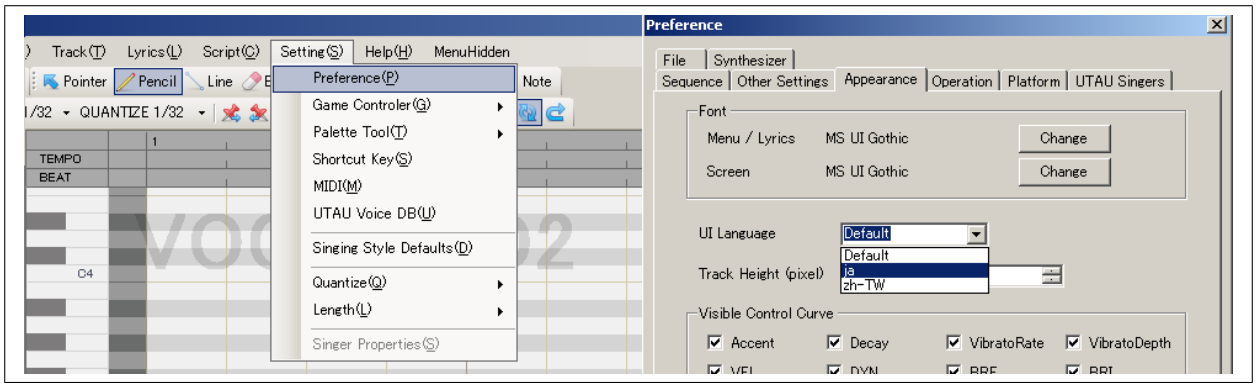
[3] 2010/11/04 現在 STAND を含んだバージョンは未リリースだよ！

3　使い方

v.Connect-STAND を Cadencii 上で使うためにはいくつかの準備が必要です。以下に挙げる手順は Cadencii に新しくインストールした UTAU 音源を認識させる作業です。UTAU 音源が認識されない状態だと v.Connect-STAND による合成は利用できません。

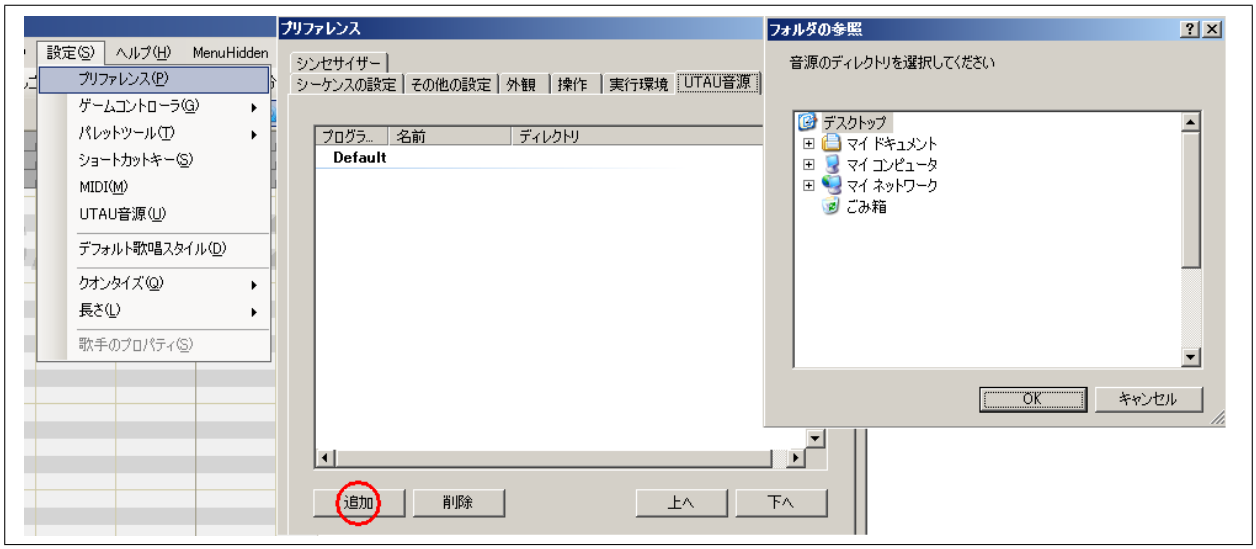
3.1　Cadencii の言語設定

Cadencii はデフォルトの言語が英語に設定されているので [Setting] → [Preference] → [Appearance] → [UI Language] から「ja」を選択してください。



3.2　UTAU 音源の登録

使用したい UTAU 音源をインストールした後 [設定] → [プリファレンス] → [UTAU 音源] → [追加] と進み、インストールした UTAU 音源のフォルダを指定してください。



UAR ファイルなど、UTAU の機能でインストールした音源は

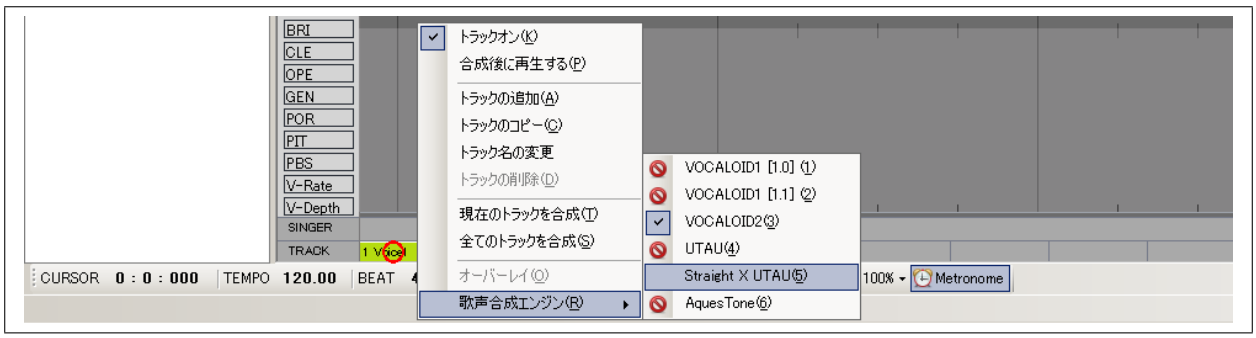
```
C:\Program Files\UTAU\voice\
```

以下にある場合が多いです。こちらは個々の環境によるので、インストール・展開したフォルダを指定してください。

なお、Cadencii は oto.ini から波形や周波数表の有無をチェックします。また charcter.txt 内に書かれている音源のキャラクター名を音源として読み込みます。charcter.txt は必ずしも必要ではないですが複数の音源の音源名が同一になった場合、Cadencii が扱う xvsq ファイルの制約上誤動作が起こる可能性があります。

3.3 合成エンジンの指定

次に Cadencii 上のトラックの合成エンジンを v.Connect-STAND に変更します。下部に表示されているトラック名を右クリック→ [合成エンジン] → [Straight × UTAU] に変更してください。



なお、[Straight × UTAU] モードが選択できなくなっている場合は

- Cadencii フォルダ内に vConnect.exe が存在しない。
- 前節の UTAU 音源の登録を行っていない

などが考えられます。UTAU 音源を登録していない場合は、UTAU モード・Straight × UTAU モードとも選択できなくなります。

また複数の UTAU 音源を登録した場合トラック名が置かれる TRACK 欄の上にある SINGER 欄をダブルクリックして歌手の変更[4]を行ってください。

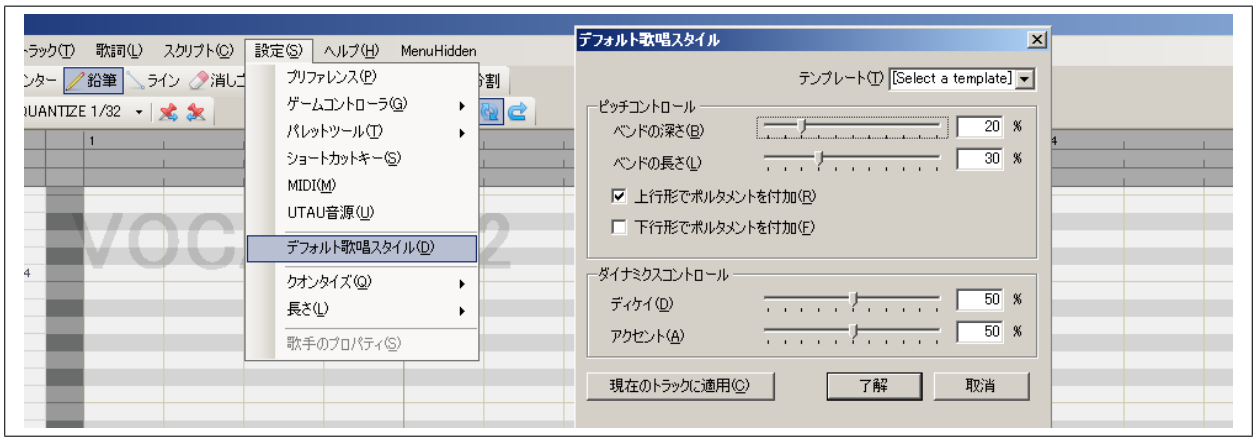
3.4 デフォルト歌唱スタイルの設定

v.Connect-STAND は様々なパラメータを使用して歌声を合成します。そのうち音符単位で設定できるパラメータ[5]の初期値を設定します。この設定が行われない場合不自然な音声が合成される場合があります。

[4] 2010/11/04 現在 v.Connect の音符の途中での歌手変更は未実装。歌手変更自体は可能だが別のシーケンスになる、んだっけ？

[5] 4 章 4.2 節にて後述

シーケンスを入力する前、あるいは Cadencii 以外のシーケンサで作成したシーケンスを読み込んだ後、[設定] → [デフォルト歌唱スタイル] を選択して設定を行ってください。特に外部のシーケンサで作成したデータを読み込む際に歌唱スタイルが適用されない[6]ので、インポートしたトラックごとにデフォルト歌唱スタイルを適用してください。



v.Connect ではポルタメント長の最短長を設定していないため、

ベンドの深さ 20〜30%

ベンドの長さ 20〜30%

ディケイ 50%

アクセント 50%

上記程度に設定しておくと合成結果がスムーズになる場合が多いです。テンポやリズムによってはベンドの長さを調節した方がよい結果になる場合もあります。これらのパラメータの働きについては 4 章 4.2 節にて後述します。

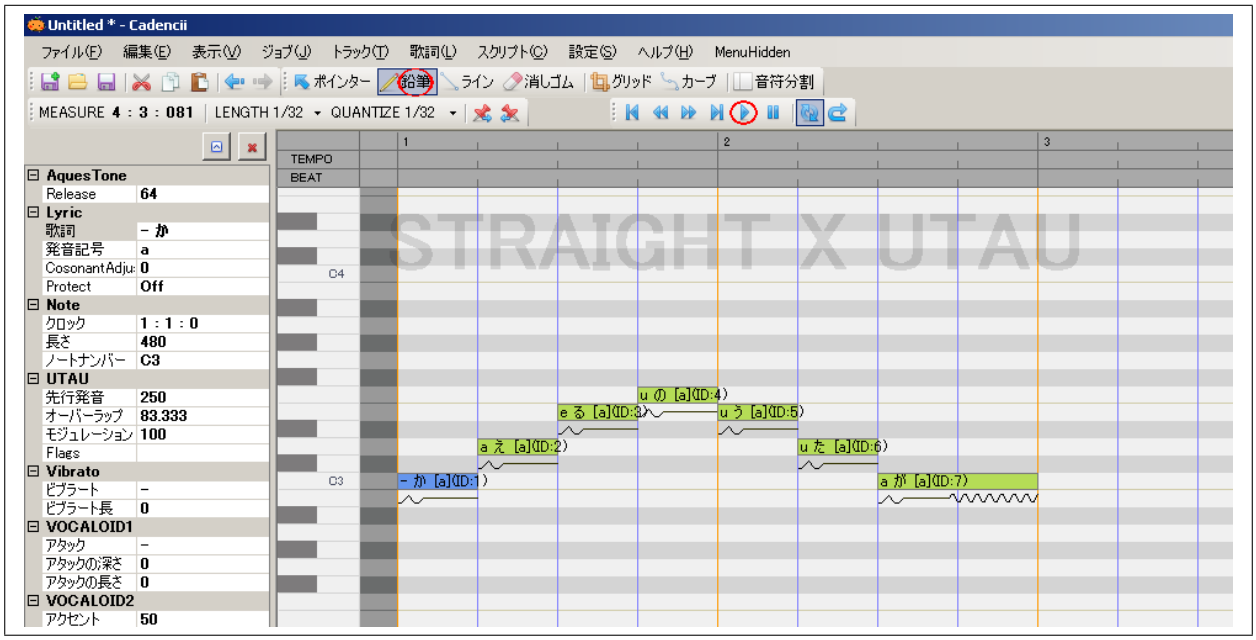
なお [現在のトラックに適用] を選択すれば設定が現在のトラック上の全ての音符に対して適用されます。MIDI など他のシーケンサで作成したデータではこれらの値は 0%として読み込まれますので、Cadencii 以外で作成したシーケンスを使用する場合は読み込み毎に各トラックに対して設定を適用してください。

3.5 シーケンスの入力

鉛筆ツールを選択した状態でピアノロール部をドラッグすると音符が作成されます。ダブルクリックするか、音符が選択された状態で [tab] キーを押すと歌詞を変更できます。

音符を選択した状態では左側のプロパティウィンドウに音符のプロパティが表示されます。

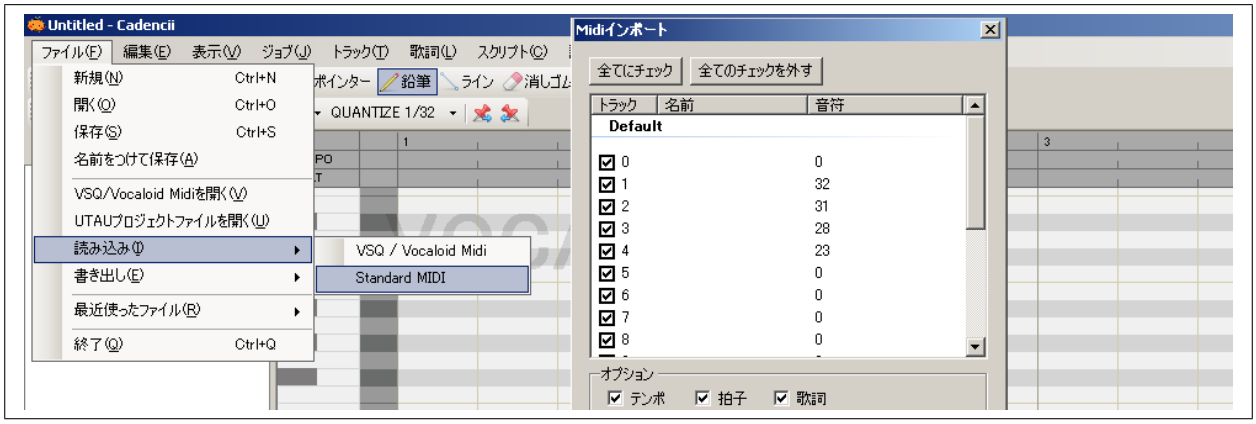
[6] 2010/11/04 現在。ほんとか？



音符の下に表示される山型のアイコンをダブルクリックすると個々の音符の表情プロパティを、平坦な部分をダブルクリックすると個々のビブラートの長さを設定できます。また、プロパティウィンドウに数値を入力しても個別の音符に変更が反映されます。

3.6 MIDI や VSQ の読み込み

Cadencii は標準 MIDI ファイルや VSQ（VOCALOID,VOCALOID2 用シーケンス）の読み込み機能があります。[ファイル] → [読み込み] から VSQ であれば [VSQ / Vocaloid MIDI] を、標準 MIDI ファイルであれば [Standard MIDI] を選択しファイルを読み込んでください。



すると MIDI インポートウィンドウが表示されますのでそこから読み込みたいデータをチェックすれば、新しいトラックに読み込んだファイルの内容が書き込まれます。この新

しいトラックではデフォルト歌唱スタイルが適用されない[7]ので、デフォルト歌唱スタイルを前節にしたがって適用してください。

[7] 2010/11/04 現在。確認ミスだったらごみん。

4　歌声制御パラメータ

4.1　歌詞

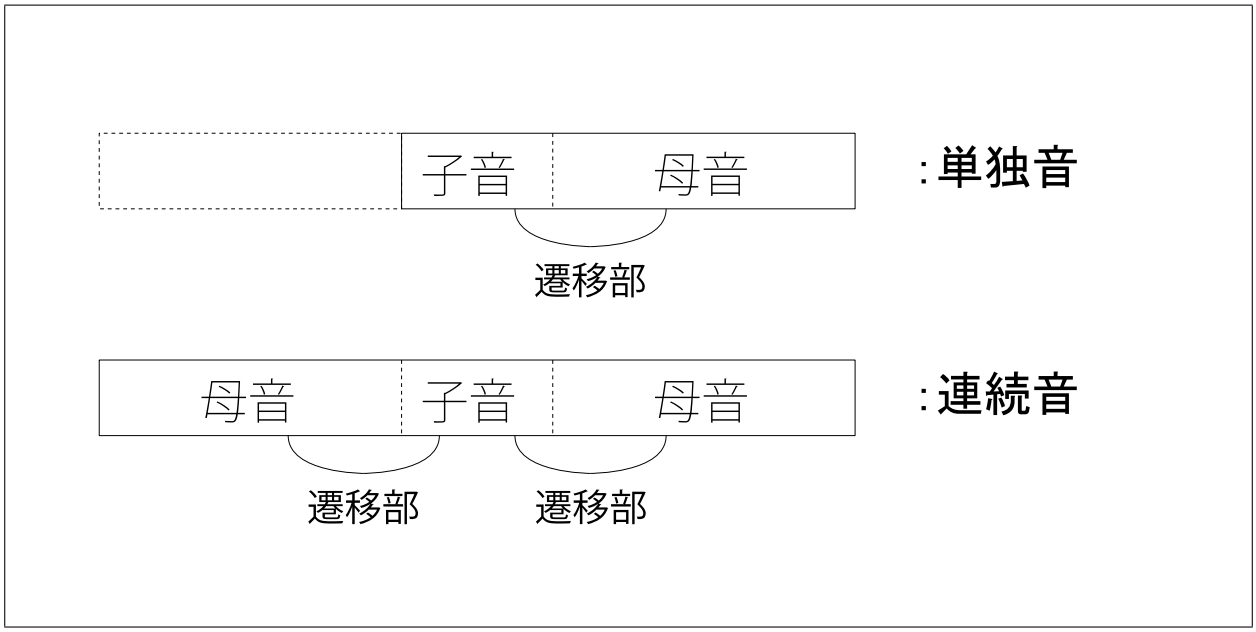
v.Connect-STAND は歌詞を認識するために UTAU 音源の原音設定ファイル（oto.ini）を参照します。v.Connect-STAND が認識できる歌詞は原音設定内のエイリアスのみです[8]。

UTAU 音源には大まかに分けて二つの形式があります。

単独音 五十音表をそのまま録音したもの。ex.)「あ」や「か」など

連続音 五十音表を組み合わせ、遷移部まで録音した形式。ex.)「a あ」や「a か」など

連続音は先行音の母音も含めて録音する形式で図示すると以下のようになります。



単独音か連続音により歌詞の表記法が異なります。

音源にもよりますがインストールした音源が単独音音源の場合は歌詞どおりに平仮名を入力する場合が多いです。「かえるのうたが」という歌詞なら「か」「え」「る」「の」「う」「た」「が」と入力すればよいでしょう。

連続音音源の場合は、先行する音符の母音を引きずった形で歌詞が表されます。「かえるのうたが」という歌詞なら「- か」「a え」「e る」「u の」「o う」「u た」「a が」になります。入力を簡単にする UTAU 用プラグイン[9] は Cadencii 上で使用可能なのでそちらを利用することもできます。

[8]辞書機能は実装する予定がありません。したとしてもずっと先の話になると思います。

[9]「UTAU ユーザー互助会 Wiki@ウィキ - プラグイン」http://www20.atwiki.jp/utaou/pages/36.html こちらに歌詞を連続音化プラグインがいくつか公開されています。

4.2　音符の表情プロパティ

音符の表情プロパティでは音符一つ一つの細かな表情を操作できます。大まかに分けて

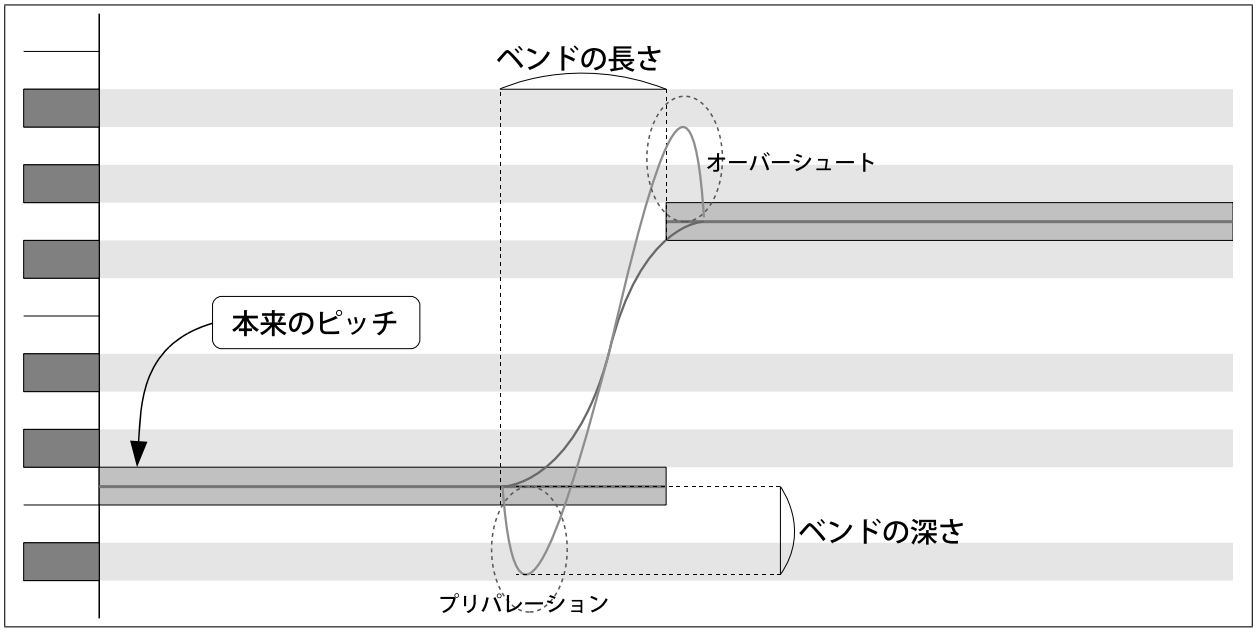
- ベンドの深さ・長さ
- アクセント・ディケイ
- ビブラート

の三種類を操作することで手軽[10]に歌声を操作できます。

音符の表情プロパティを編集する際は、ピアノロール上の音符を右クリック→[プロパティ] を選択してください。

4.2.1　ベンドの深さ・長さ

人間の声は音符の音高に変化があった場合先行音と後続音の間にポルタメントと呼ばれる滑らかなピッチ変化が起こります。v.Connect-STAND ではこの現象をベンドの深さ・長さによって再現します。



プリパレーションとはピッチが動き始める前に逆方向に動く現象で、オーバーシュートはピッチが安定する前に行き過ぎる現象です。ベンドの長さは音符の長さの%指定そのものです。ベンドの深さはオーバーシュート・プリパレーションの深さを、後続音との音程の%指定で行います。つまり、ベンドの深さ・音程の差が大きいほどプリパレーション・オーバーシュートは大きくなります。

[10]全てのピッチを手書きするよりは、程度と考えてください。

4.2.2　音量プロパティ

v.Connect-STAND はディケイ・アクセントにより個別の音符間の音量を操作できます。大まかに分けてディケイは音符の定常部の音量、アクセントは音符のアタック部の音量を変更します。

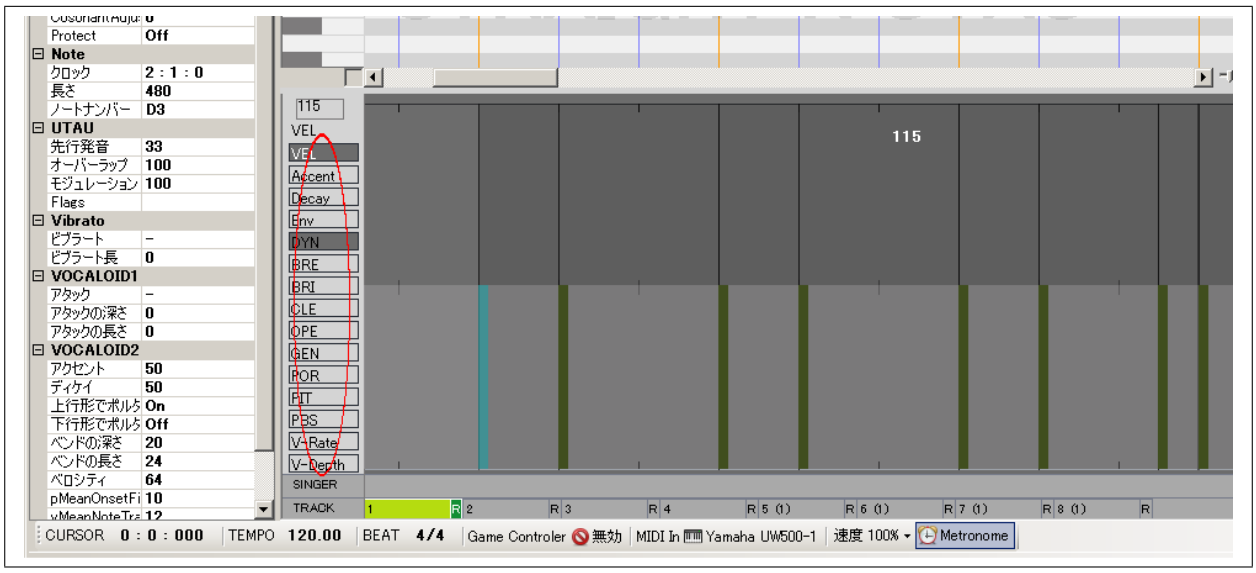
なお、UTAU の原音は固定長終了位置の音量によって音量をそろえてから使用いたします。そのため個別の音符の音量を考えずともある程度音量の揃った状態で歌声が出力されます。

4.2.3　ビブラート

Cadencii 上の音符の表情プロパティからはビブラートのかかる長さのみ変更できます。細かな設定は次節のコントロールトラックから編集できるのでそちらを参照してください。

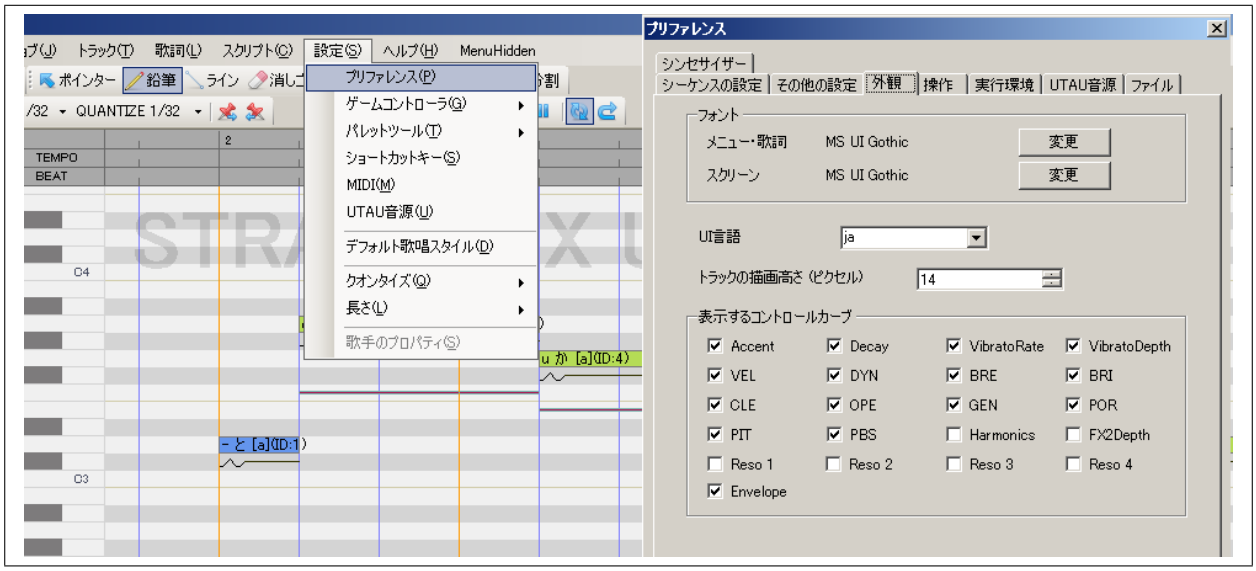
4.3　コントロールトラック

コントロールトラックは Cadencii 上ではピアノロール下部にて編集できます。



なお、表示したいコントロールを増やしたり減らしたりしたいときは [プリファレンス] → [外観] → [表示するコントロール] から該当する項目のチェックを変更してください。

コントロールトラックは VEL/DYN/PIT/PBS/GEN/BRI/CLE に対応しています。UTAU 音源を扱う都合上エンベロープのうち先行発音・オーバーラップにも対応しています。また、ビブラートは VOCALOID2 Editor とは異なりコントロールトラック上でポルタメントの速さ・深さを編集します。ビブラートの細かな編集をしたい場合は、VibratoRate・VibratoDepth のチェック欄にチェックをつけてください。



4.3.1 VEL

VOCALOID2 及び v.Connect-STAND において VEL は子音速度を表します。実装自体は大きく違うので効果のほどは違いますが、おおむね該当する音符のアタック部分までが変化します。デフォルト値は 64 で値を 0 にすると子音速度が 0.5 倍・子音長が 2 倍となり、値を 128 にすると子音速度が 2 倍・子音長が 0.5 倍になります。

当ツールでは発音の遷移部の速度がそのまま変化いたしますので、舌っ足らずになるなど遷移部分の違和感は VEL で多少軽減できる可能性があります。

4.3.2 DYN

振幅を操作します。デフォルト値は 64 で単純に 0〜127 の間で振幅を変更します。

4.3.3 PIT/PBS

ピッチを変化させます。PBS で指定した量が最大変化幅になり、PIT でその幅内を変化させます。PBS は 0〜12・PIT は-8192〜8191 の間で変化しデフォルト値はそれぞれ 2,0 です。

4.3.4 GEN

スペクトルを線形に伸縮します。GEN を下げるとテープを早く回したときのような声になり、上げると逆にテープを遅く回したときのような声になります。0〜127 まで変化でき、デフォルト値は 64 です。

4.3.5 BRI/CLE

BRI/CLE は Voice Texturing 技術を用いて声色を変化させるパラメータです[11]。内部で使用される形式がまだ未定なので、7.5 章で説明はするものの開発中につき仕様が著しく変更される場合があります。

4.3.6 エンベロープ

v.Connect-STAND は発音位置を音符の位置及びエンベロープから計算します。ここで変更できる値は先行発音とオーバーラップでそれぞれ

- 先行発音：発音位置を音符位置から指定された量だけ前にずらします。
- オーバーラップ： 該当する音符が単独音である場合、指定された量だけ該当する音符に先行する音符の長さを長くします。

という機能を持ちます。指定はミリ秒単位です。Cadencii は歌詞を入力した際に該当する UTAU 原音設定をエンベロープの値に設定します。

[11] そうなったらいいなぁ。実装見直し中のため処理を凍結中です。

5 トラブルシューティング

5.1 Cadencii が起動しない

- 2.3 節に従って.NET Framework 2.0 以上のランタイムをインストールしてください。

5.2 v.Connect-STAND を合成エンジンに指定できない

- UTAU 音源が登録されているか確認してください。未登録の場合 3.2 節に従って UTAU 音源を登録してください。

5.3 音が鳴らない

- 2.3 節に従って VisualC++ランタイムをインストールしてください。なお、Cadencii フォルダ内の vConnect.exe をダブルクリックしたときにエラーダイアログが表示されなければ正しくインストールされています。
- UTAU 音源中に存在する歌詞か確認してください。UTAU 音源内の oto.ini 中の歌詞のみ使用可能です。
- DYN の値やアクセント／ディケイの値が正しく設定されているか確認してください。

5.4 発音がおかしい

- 各音符の先行発音が正しく設定されているか確認してください。なお、Cadencii 上で先行発音などが設定されるタイミングは音符に歌詞が入力されるタイミングです。歌詞を変更するプラグインを使用した場合も同様です。
- VEL の値が正しく設定されているか確認してください。デフォルト値は 64 でそれ以上だと発音が短く・それ以下だと発音が長くなります。
- アクセント／ディケイの値が正しく設定されているか確認してください。

5.5 音程が滑らかに繋がらない

- ベンドの深さ／長さが正しく設定されているか確認してください。特にベンドの長さが極端に短い場合は音の繋がりが不自然になることがあります。

5.6　音量の遷移が不自然

- アクセント／ディケイの値が正しく設定されているか確認してください。
- v.Connect-STAND は音源中の波形データを使用する前に音量を揃える動作を行います。発音によって音量の小さな音素があった場合接続が不自然になる場合があります。

6　コマンドライン

v.Connect-STAND はコマンドラインアプリケーションです。コマンドラインから以下のような形で使用が可能[12]です。

```
vConnect [vsq_meta_text_path] [out_wave_path]
```

あるいは

```
vConnect -i [vsq_meta_text_path] -o [out_wave_path] {options}
```

の形でコマンドラインから合成が可能です。

また WORLD のスペクトルを出力するオプションを使用すれば、

```
vConnect -w -i [utau_oto_ini_path] -l [alias_to_analyze] -o [out_wsp_path] {options}
```

の形で指定した音素の WORLD 分析結果を wsp 形式[13]で出力できます。

6.1　オプション

v.Connect-STAND が受け付けるオプションの種類と意味は以下の通りです。

```
-i [path]                          path of input-file
-o [path]                          path of output-file
-s                                 slow synthesize mode
-nf0                               no spectral transform by f0
-nvn                               no volume normalization
-w                                 wsp output mode
-l [alias]                         alias of oto.ini with -w option
-charset-otoini [charset-name]     charset of oto.ini (default: Shift_JIS)
-charset-vxt [charset-name]        charset of input-file (default: Shift_JIS)
-list-charset                      print supported charset list
```

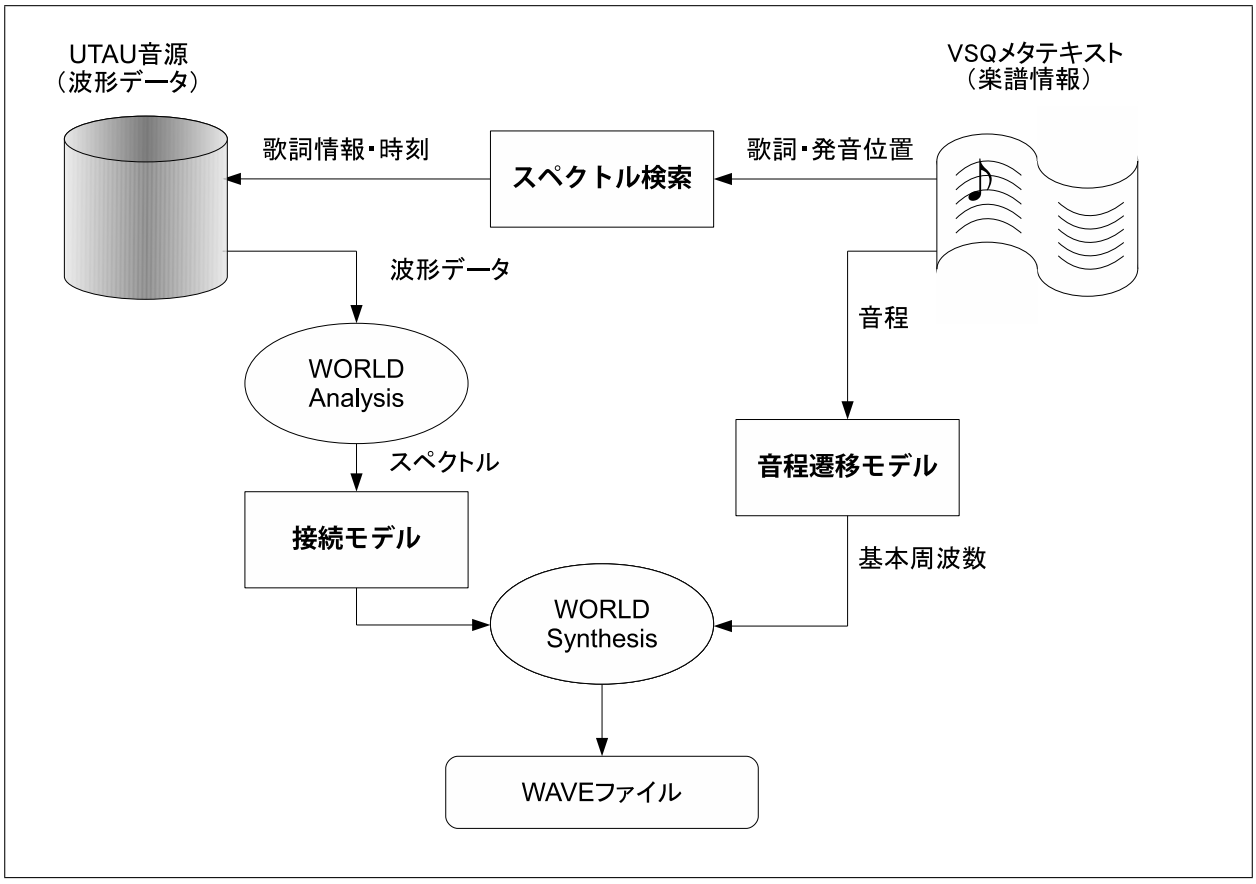
なおコマンドラインオプション等の仕様は実装により頻繁に変化する場合があります。最新版でのオプションは souceforge 上のソースコードを参照してください。

[12]あまりお勧めはしません。
[13]詳細未定。現在はアスキー文字で各種情報を列記。

7　付録資料

7.1　動作の流れ

v.Connect-STAND は Cadencii から拡張 VSQ メタテキストを受け取り、それをもとに UTAU 音源から波形を取り出し WORLD にスペクトルを計算する。加えて、VSQ メタテキスト内のピッチ情報から基本周波数を計算し、適切な位置のスペクトルを取り出しピッチ情報とあわせて WORLD に再合成させることにより音声を得る。



WORLD は声の特徴[14]とピッチに分ける、あるいは声の特徴とピッチから音声を合成する Vocoder ベースの音声分析合成系である。v.Connect-STAND は VSQ メタテキストをもとに UTAU 音源から WORLD を使用して音声の分析・再合成を自動的に行うように設計されている。

[14]厳密には STAR スペクトルと PLATINUM 非周期性指標

7.2 スペクトル接続規則

v.Connect-STAND はシーケンス内に記述された値から、その時刻に適した STAR スペクトルを検索する。その際発音位置や音符長は vsq 拡張メタテキスト内に記述された先行発音・オーバーラップ・歌詞・ベロシティから適宜計算され、その値を使用し必要であれば接続時に対数スペクトルでの加減算を行う。

7.2.1 発音位置

発音位置は音符の位置 t_{note}[s]，ベロシティ v[-]，先行発音長 l_{pre}[s] によって決定される。ここで、実際の発音位置 t_{actual} は

$$t_{\mathrm{actual}} = t_{\mathrm{note}} - l_{\mathrm{pre}} \times 2^{\frac{(64-v)}{64}} \tag{1}$$

で与えられる。ベロシティは 0〜127 の値をとるので、7.2.2 節と合わせて子音部分の長さが 2〜$\frac{1}{2}$ 倍の間で変化する。

7.2.2 検索位置

UTAU 音源内の波形ファイルは歌詞情報と原音設定から切り出して使用する。その際ベロシティによって時間伸縮を行う。n 番目の先行発音による補正後の発音位置が t_{actual}[s] でシーケンス内の時刻 t[s] のときに切り出した波形内での時刻 $T_n(t)$[s] は、UTAU 原音設定内の固定長 l_{fixed}[s] として

$$T_n(t) = \begin{cases} (t - t_{\mathrm{actual}}) \times \dfrac{1}{2^{\frac{(64-v)}{64}}}, & T_n(t) \leq l_{fixed} \\ l_{\mathrm{fixed}}, & \text{else.} \end{cases} \tag{2}$$

で与えられる。

7.2.3 発音長

v.Connect-STAND では発音長は音符の長さ・先行発音・オーバーラップおよび歌詞によって決定される[15][16]。

まず n 番目の音符の長さを l_{tick}[s] として仮の長さ $\tilde{l_n}$[s] を以下のようにする。

$$\tilde{l_n} = l_{\mathrm{tick}} + l_{\mathrm{pre}} \times 2^{\frac{(64-v)}{64}} \tag{3}$$

このとき、実際の長さ l_n[s] は $n+1$ 番目の発音位置 t_{n+1} と歌詞が連続音かどうかのブール値 isVCV により

$$l_n = \begin{cases} \tilde{l_n}, & \mathrm{isVCV} = \mathrm{true} \quad or \quad t_n + \tilde{l_n} < t_{n+1} \\ t_{n+1} - t_n + l_{\mathrm{overlap}}, & \text{else.} \end{cases} \tag{4}$$

で与えられる。ここで、l_{overlap} は $n+1$ 番目のノートが持つオーバーラップの値である。

[15] 直感的に言ってしまえば、先行発音分の調整を行っています。
[16] オーバーラップが考慮されるのは先行音と近接した単独音歌詞をもつ音符のみです。

7.2.4 スペクトル

v.Connect-STAND は以上の副節で設定した値を用いて各波形から STAR スペクトルを計算し、対数スペクトル上での加減算を行って合成時に使用するスペクトルを得ます。

7.2.2 節で得られた元波形内での時刻を元に、n 番目と n+1 番目の時刻 t に対応する STAR スペクトル $X_n(T_n(t))$, $X_{n+1}(T_{n+1}(t))$ を検索する。これらに対して、合成に使用するスペクトル $X(t)$ は

$$\log X(t) = \begin{cases} r(t)\log X_n + (1-r(t))\log X_{n+1}, & t \leq t_n + l_n \ \ and \ \ t_{n+1} \leq t \\ \log X_n, & \text{else.} \end{cases} \tag{5}$$

で与えられる。このとき $r(t)$ は

$$r(t) = \begin{cases} \frac{t}{1-t} \\ 1 \quad \text{else.} \end{cases} \tag{6}$$

により与えられる。

直感的に言えば、副節で求めた発音位置・発音長を基に後続音と重なる部分があれば滑らかに接続する処理を行っている。

7.3 音程遷移規則

v.Connect-STAND の音程遷移規則は SingBySpeaking[8] や MandarinSynthesis[9] などを参考に、よりシンセサイザーとして分かりやすい形にして実装している。

歌声音声中の基本周波数の変化は

1. ポルタメント
2. プリパレーション
3. オーバーシュート
4. ビブラート
5. 微細振動

に分けて考えられることが多く、このうち１～４は一般的に伝達関数が $H(s) = \dfrac{k}{s^2 + 2\zeta\omega s + \omega^2}$ のフィルタで考える場合が多いが、v.Connect-STAND では処理量と合成結果を鑑みた結果、近似式を用いている[17]。

7.3.1 ポルタメント

現在の音符と次の音符の周波数をそれぞれ f_1 [Hz]、f_2 [Hz] とし、現在の音符の開始位置を t_1 [s]、現在の音符長を l_1 [s]、現在の音符長に対するベンド長の割合を r_1 [-] $(0 \leq r_1 \leq 1)$ としたとき、ポルタメントを考慮した周波数 f_{por} [Hz] は以下のようになる。

$$f_{\mathrm{por}}(t) = f_1 + \frac{1}{2}(1 - \cos \pi x)(f_2 - f_1), t_1 \leq t \leq t_1 + l_1. \tag{7}$$

ただし、

$$x(t) = \begin{cases} \dfrac{t - \{t_1 + (1 - r_1)l_1\}}{r_1 l_1}, & t_1 + (1 - r_1)l_1 \leq t \leq t_1 + l_1, \\ 0, & \text{else.} \end{cases} \tag{8}$$

より直感的に言ってしまえば 7.2 章の接続規則で計算した発音位置・長さをもとに、元の音符長のうちベンドの長さでパーセント指定された長さだけ滑らかに接続している (図 1 参照)。

7.3.2 プリパレーション・オーバーシュート

プリパレーションとオーバーシュートを適用した周波数 f_{dep} は

$$f_{\mathrm{dep}}(t) = \begin{cases} f_{\mathrm{por}}(t) \times pow(f_1 - f_2,\ \sin(2\pi \frac{t - t_2}{r_1 l_2})), & |t - t_2| \leq r_1 l_1, \\ f_{\mathrm{por}}(t), & \text{else.} \end{cases} \tag{9}$$

で与えられる。ポルタメントを中心に減衰振動様のピッチ変動を加えることで実装している。この 2 要素によるピッチ変動の長さはポルタメントの長さ、つまり先行音における音符の長さにベンドの長さをかけたもの、により決定され、ピッチ変動の大きさはベンドの深さ分だけ音符間の周波数の差を乗じたものとなる。

[17] 伝達関数が時間変化して都合が悪いのに加えて、合成結果にさほど差が無かった

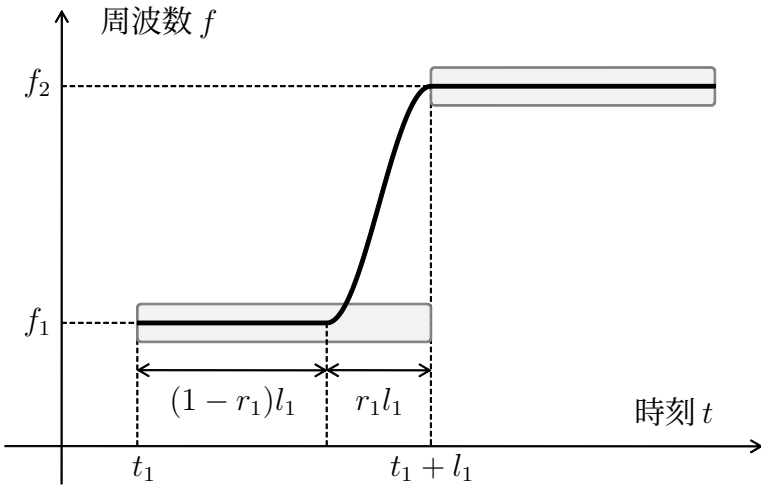


図 1: ポルタメント

7.3.3 ビブラート

7.3.4 微細振動

7.4 音量遷移規則

v.Connect-STAND では音量の調節は大まかに WORLD による分析合成の前後で二つに分けられる。すなわち

1. 分析時の正規化
2. 音量遷移規則による音量曲線の編集

の二つである。

また、音量遷移規則による音量曲線の編集段階では合成後に歌声の表情としての音量変化を付加させる。三種類の音量遷移規則が存在し、それぞれ

1. DYN パラメータによる音量操作
2. Accent/Decay パラメータによる音量操作
3. 旋律末尾における語尾処理

である。ここでは振幅増幅関数 $a(t)$ を使用する。WORLD による合成後の音声 $\tilde{y}(t)$ に対し、実際に v.Connect-STAND が出力する音声 $y(t)$ は、

$$y(t) = \tilde{y}(t) \times a(t) \tag{10}$$

で与えられる。ただし DYN による音量変化を $a_{\mathrm{dyn}}(t)$、Accent/Decay による音量変化を $a_{\mathrm{acc}}(t)$、音符末尾における語尾処理による音量変化を $a_{\mathrm{tail}}(t)$ としたときに、

$$a(t) = a_{\mathrm{dyn}}(t) \times a_{\mathrm{acc}}(t) \times a_{\mathrm{tail}}(t) \tag{11}$$

で a(t) は与えられる[18]。各関数は後節で定義するものとする。

7.4.1 分析時の正規化

UTAU 音源内には音量のばらつきが存在する。音量のばらつきを最低限のものとするために原音内の音量を用いてあらかじめ波形を正規化する。

まず、時刻 t における音量 A を以下の式で定める。

$$A(t) = \sqrt{\frac{1}{2N} \sum_{i=-N}^{N} |h(i) \times x(t+i)|^2} \tag{12}$$

ここで $x(t)$ は波形とし、$h(n)$ は $2N$ の長さのハニング窓とする。音量をこのように定義し UTAU 音源から切り出した波形の開始時刻を t_{begin}、固定長終了位置を t_{cend} としたとき、

$$v = \max(\ A(t_{\mathrm{begin}}),\quad A(t_{\mathrm{cend}})\ ) \tag{13}$$

[18] $a(t)$ は実処理中では 1[ms] 単位か 2[ms] 単位で与えられるので間の値は線形補間される。

なる v を正規化に使う音量とする。ここで $\max(x, y)$ は $x \leq y$ で y を、それ以外で x を返す関数である[19]。ここで、

$$\tilde{x}(t) = \frac{x(t)}{v} \tag{14}$$

なる $\tilde{x}(t)$ を使用し WORLD により合成に使用するパラメータを分析させる。詳細は参考文献 [1][2] を参照のこと。

7.4.2 DYN パラメータによる音量操作

時刻 t における DYN の値を $d(t)$ としたとき、DYN による振幅増幅関数 $a_{\mathrm{dyn}}(t)$ は

$$a_{\mathrm{dyn}}(t) = \frac{d(t)}{64} \tag{15}$$

で与えられる。

7.4.3 Accent/Decay パラメータによる音量操作

暫定実装につき詳細は割愛。アタック位置を固定長終了位置、ディケイ位置を固定長終了位置と音符終了位置との中点とし間を余弦関数を使用して滑らかに繋ぐ関数としている。

7.4.4 旋律末尾における語尾処理

v.Connect-STAND は後続音を持たない音符に対しては旋律末尾だと解釈し音量による語尾処理を行う。前節で求めた該当音符の終了時刻 t_{end} を用いて、

$$a_{\mathrm{tail}}(t) = \begin{cases} \frac{|t_{\mathrm{end}} - t|}{c}, & 0 < t_{\mathrm{end}} - t < c \\ 0, & t_{\mathrm{end}} - t \leq 0 \\ 1, & \text{else} \end{cases} \tag{16}$$

ここで c は定数で暫定的に 50[ms] としている[20]。

[19] 要は頭と固定長終了時の大きい方で正規化すれば問題が少ないだろうと言う打算。
[20] 違和感を減じるためだけの処理です。

7.5　周波数変換

ただいま実装を変更中。

参考文献

[1] 森勢将雅, 河原英紀, 西浦敬信; “基本波検出に基づく高 SNR の音声を対象とした高速な F0 推定法,” 電子情報通信学会 論文誌 *D*, **J93-D**, 109-117 (2010)

[2] 森勢将雅, 中野皓太, 西浦敬信; “歌唱合成システムの実現を目的とした高品質音声分析合成法の提案,” *IEICE technical report*, **110**(71), 89-94 (2010)

[3] 飴屋／菖蒲; “歌声合成ツール UTAU サポートページ,” http://utau2008.web.fc2.com/

[4] Bonada, J., Serra, X.; “Synthesis of the Singing Voice by Performance Sampling and Spectral Model,” *IEEE Signal Processing Magazine*, **24**(2), 67-79 (2007)

[5] kbinani; “cadencii_jp @ wiki - Cadencii,” http://www9.atwiki.jp/boare/pages/18.html

[6] Matteo Frigo, Steven G. Johnson; “FFTW Home Page” http://www.fftw.org/

[7] Free Software Foundation, Inc.; “libiconv - GNU Project - Free Software Foundation,” http://www.gnu.org/software/libiconv/

[8] 齋藤毅, 後藤真孝, 鵜木祐史, 赤木正人; “SingBySpeaking：歌声知覚に重要な音響特徴を制御して話声を歌声に変換するシステム,” 情報処理学会研究報告, 2008-MUS-74-5

[9] Shu-Sen Zhou, Qing-Cai Chen, Dan-Dan Wang, Xiao-Hong Yang; “A corpus-based concatenative Mandarin singing voice synthesis system,” *Machine Learning and Cybernetics*, *2008 International Conference on*, **5**, 2695-2699 (2008)