

|                     |                                                                                 |
|---------------------|---------------------------------------------------------------------------------|
| <b>Title</b>        | 音声の韻律モデルに基づく基本周波数パターンの再構成                                                       |
| <b>Author(s)</b>    | 隈田, 章寛                                                                          |
| <b>Citation</b>     |                                                                                 |
| <b>Issue Date</b>   | 1997-03                                                                         |
| <b>Type</b>         | Thesis or Dissertation                                                          |
| <b>Text version</b> | author                                                                          |
| <b>URL</b>          | <a href="http://hdl.handle.net/10119/1028">http://hdl.handle.net/10119/1028</a> |
| <b>Rights</b>       |                                                                                 |
| <b>Description</b>  | Supervisor:木村 正行, 情報科学研究科, 修士                                                   |

# 音声の韻律モデルに基づく 基本周波数パターンの再構成

隈田 章寛

北陸先端科学技術大学院大学 情報科学研究科

1997 年 2 月 14 日

**キーワード:** ピッチパターン,  $F_0$  生成過程モデル, 再構成, フィルタ, ビーム探索

## 1 はじめに

韻律情報の分析にはピッチ周波数パターンを用いる事が多く、それゆえに、より正確なピッチ抽出が望まれている。ピッチパターンを認識に利用するには、連続で滑らかであるほうが都合が良いため、しばしば平滑化の処理が行なわれる。平滑化には線形補間などの手法が用いられるが、いずれも数値解析的な処理であり、これらの平滑化は韻律制御の物理的な特徴を表していない。そこで本研究では、ピッチパターンを生成機構に則したモデルを用いて再構成する手法を提案する。

## 2 $F_0$ 生成過程モデル

ピッチパターン生成の物理的特徴を考慮にいたした  $F_0$  生成過程モデルを用いる。このモデルではピッチパターンは句頭から句末にかけて緩やかに下降するフレーズ成分と、局所的に起伏するアクセント句に対応するアクセント成分との和として捉えられる。フレーズ、アクセント各成分の逆フィルタを表す伝達関数を求めることで、ピッチ周波数パターンからそれぞれの指令のタイミング、ならびに大きさを検出する逆フィルタが作成できる。

## 3 ピッチパターン再構成法の概要

既存のピッチ自動抽出法を用いて得られたピッチパターンを  $F_0$  生成過程の逆フィルタに通すことにより、モデルの指令を推定する。これにより任意の時刻においてフレーズあるいはアクセントの指令が発生したと仮定した場合の指令の大きさが得られる。これらの推定された指令による全ての組合せの指令系列仮説のうち最も再構成歪みの小さいものを探す。しかし、全ての仮説について再構成を試行することは膨大な計算量を要するので、時間同期型のビーム探索を行うことにする。

### 3.1 指令候補の推定

はじめに入力ピッチパターンから、フレーズ指令、アクセント指令をそれぞれの制御機構の逆フィルタを用いて検出する。ただし、逆フィルタに入力するピッチパターンによって2種類の手法を試す。

**自動抽出ピッチパターンを入力とする方法** 自動抽出ピッチパターンをそのまま逆フィルタの入力とする方法。この方法では理想的な（例えば、生成モデルから生成された）パターンでは指令の推定精度が良いが、観測パターンではピッチの抽出誤り、揺らぎ、量子化誤差があるため推定精度が悪い。また、指令間隔が近接している箇所では、先行する指令の成分が大きく影響して、当該時刻における正しい指令の大きさを推定できないという欠点がある。

**過去の指令成分を除去したパターンを入力とする方法** 処理中の時刻までに推定した指令系列候補によって生成される成分を観測ピッチパターンから除き、その差分パターンを逆フィルタへ入力する。これにより、各指令の成分どうしの重なりによる推定誤りを避けることが出来る。

しかし、逆フィルタでは指令の発生時刻のピッチ抽出が正確でなければ正しい指令が推定できない。そこで、逆フィルタ以外のフィルタについても検討する。

**基本指令成分を用いたフィルタ** 各指令によって生成される成分は、基本指令成分上の各時刻のピッチを指令の大きさで掛けたものに等しい。よって、自動抽出ピッチパターンを基本指令成分で割ることで、指令の大きさを推定できる。この方法ならば、指令の発生時刻のピッチが不正確でもそれよりも後のピッチで指令の推定が可能になる。

### 3.2 時間同期型の指令探索

発声開始時刻から処理中の時刻  $t$  までのパターンに対して二乗誤差歪みを小さくする最適な  $M$  個の系列が推定されていると仮定する。ここで、指令系列の候補数をビーム幅として定義し、 $M$  はその最大値である。逆フィルタにより、フレーズ指令  $C_p$ 、アクセント指令  $C_a$  が推定され、 $C_a$  が正の時はアクセント開始指令、負の時はアクセント終了指令となる。時刻  $t-1$  における  $M$  個の指令系列候補の中から当該時刻に指令  $C_p$  が発生し得る指令系列を取り出し、歪み  $D_{m,t}$  が小さい順に最大  $N$  個（n-best）を選択し、指令  $C_p$  を加える。 $C_a$  についても同様の処理を行う。また当該時刻に指令が何も起こらない仮説として、時刻  $t-1$  の系列候補の中から歪みの小さい  $M-2N$  個の系列を残し、常に最大  $M$  個の仮説を保つ。この処理を発声終了時刻まで行う。

### 3.3 ピッチパターンの再構成

最終時刻における1位の指令系列をもとにピッチパターンを再構成する。

## 4 理想的ピッチパターンとの歪みによる評価

各手法によって再構成されたピッチパターンと A-b-S (Analysis by Synthesis) によって得られた理想的ピッチパターンとの歪みを計算することで評価を行った。比較のため、平

滑化の処理をしない自動抽出ピッチ、従来法として線形補間、移動平均、直線近似の手法についても実験した。

## 5 再構成パターンを用いた句境界検出

これらの再構成結果を用いて、 $F_0$ パターン連続整合法により句境界検出をおこなった。比較のため、平滑化の処理をしない自動抽出ピッチ、従来法として線形補間、移動平均、直線近似の手法についても実験した。

## 6 結果

再構成を行ったところ、部分的に観測ピッチパターンの物理的な特徴が表現されていない箇所があるが、全体的に近似した形が得られることがわかった。

また、ビーム幅  $M$  と  $n$ -best 数  $N$  をそれぞれ変化させた場合の歪みの変化を調べた。この結果から、一般にビーム幅を大きくするにつれて歪みが小さくなることが確認でき、一定のビーム幅に対しては歪みを最小にする最適な  $n$ -best が存在することがわかった。また、基本指令成分を用いた方法が最も歪みが小さく、次に過去の指令による成分を除去した方法が良好な再構成が可能であった。今回用いた手法は、歪みによる比較から従来法に比べ、ピッチ抽出誤りや無声音の部分の不正確なピッチによる影響が少ないことがわかった。また、ピッチ信頼度の閾値が不適切で本来あるべきピッチパターンの一部分が削除されていても、その前後のピッチから欠落部分を復元できた。自動で平滑化を行う場合、入力ピッチパターンの状態に左右されにくく、安定した平滑化が可能であると思われる。句境界検出の結果、本研究で用いた手法により検出精度の向上が見られた。問題点としては、いずれの手法においても、最小二乗誤差近似を重視しているため、韻律の物理的な構造上、アクセントやフレーズの指令の発生が不適当だと思われる時刻にも指令を付加してしまうという傾向がある。