

計算機によるピアノ伴奏

小川 大典 戴 岡 五十嵐 滋
筑波大学工学研究科 筑波大学理工学研究科 筑波大学電子・情報工学系

我々は10年以上にわたって PSYCHE と呼ばれる音楽情報処理研究を行ってきた。PSYCHE プロジェクトの大きな目標は、計算機によるアコースティック楽器での表情豊かな美しい演奏の実現である。

このプロジェクトの柱の一つに、実時間制御を伴うアンサンブルシステムの開発があり、これまでにいくつかのタイプのアンサンブルシステムが開発されてきた。

我々は現在、演奏の表情付けの単位となる楽曲構造を重視して様々な研究を行っている。その中から今回は、楽曲構造に基づく模倣性と再現性とを導入した自動伴奏システムと、一様な遅れを持つ生ピアノを先行制御する自動連弾システムについて報告する。

Computerized Piano Accompaniment

Daisuke Ogawa*, Gang Dai†, Shigeru Igarashi‡

*Dr. Program in Eng., Univ. of Tsukuba

†M. Program in Sci. & Eng., Univ. of Tsukuba

‡Inst. of Inf. Sci. & Electr., Univ. of Tsukuba

Psyche, a music information processing project, has been pursuing impressive and expressive performance on acoustic instruments controlled by computers for more than ten years. One of the significant systems of Psyche is a group of accompaniment systems. Real-time control is the key of the systems. Also Psyche pays attention to music structure. In this paper, two accompaniment systems are described. One simulates rehearsal performance by human players based on music structure. The other is an automated piano accompaniment system. In the latter system, we solved the problem of delay in controlling an acoustic piano.

1 はじめに

我々はピアノ等のアコースティック楽器による表情豊かな美しい演奏を目標とした、PSYCHE と呼ばれる音楽情報処理研究プロジェクトを行ってきた。

音楽における表情付けは一般にフレーズ単位で行われ、その中で演奏者の意図や特徴、演奏における作法などが表現されるため、我々は楽曲構造に注目して研究を行っている。[2]

PSYCHE プロジェクトは大別して三つの柱で構成される。一つは、実際の演奏の特徴を抽出して図形的に表現する解析、二つ目は計算機によるピアノ演奏に対する、楽曲構造に基づいた演奏表情の生成、そして三つ目は、実時間制御を伴うアンサンブルシステムの開発である。

今回は楽曲構造を重視したアンサンブルシステム開発の現状を報告する。

2 楽曲構造

楽曲を構成する基礎単位にはいくつか種類がある。

- motif 楽曲を構成する最小単位。様々な変化を伴って反復される。原則として2小節のまとまりを持つ。
- phrase 旋律の自然な区切り。原則として4小節より成り、2個の motif を含む形である。
- sentence 原則として8小節より成り、2個の phrase を含む。

3 自動伴奏システム

3.1 システムの概要

人間の独奏者の演奏に合わせて伴奏パートの演奏を制御する自動伴奏システムが我々のグループで開発されている。

しかし、既存の自動伴奏システムは直前の1音の演奏情報のみを用いて将来のテンポを予測するため、システムの演奏のテンポ変化が細かく揺らいでしまい、自然な演奏が行えない。この点を解

決するためには、楽曲構造に基いてテンポの変化をコントロールすれば良いと思われる。

そこで、楽曲構造の分析に基づく自動伴奏システムを開発した。このシステムは、曲の音楽構造情報を含む楽譜データと独奏者のリハーサルデータとの組み合わせから演奏プランを作成しておき、本番の演奏では各音楽構造の冒頭一拍のテンポをその音楽構造の基礎テンポとして演奏プランを修正し、独奏者とのアンサンブルを行う。

また、独奏者の演奏する音楽構造と類似した構造が伴奏パート内にも現れる場合は、その演奏表情は独奏者のリハーサル時の演奏表情を真似するようにした。本番演奏ではその部分の伴奏パートには自主性を持たせ、システムは独奏者のテンポを追従せずに、その前の基礎テンポで演奏プランを修正し伴奏を行う。

3.2 楽曲構造情報の与え方

例として、フレーズ情報を与える場合を説明する。通常、楽曲の中にはフレーズが複数存在している。それらを分類し番号を付け、出現回数を記述する。またフレーズ的位置は、part 番号と begin 位置と end 位置によって記述する。

phrase [phrase 番号] [回数]

part 番号 :

begin : 小節番号、拍番号

end : 小節番号、拍番号

part 番号は旋律パートと伴奏パートをそれぞれ part1 と part2 として記述する。

begin 位置は所在する小節とその小節の中の拍に対応する。(小節は順番に番号を付ける。拍は小節の中の順番で番号を付ける。)

end 位置は begin 位置と同じような方法で指定する。

3.3 リハーサルシステム

あらかじめシステムには旋律パート・伴奏パートの楽譜情報を与えておく。システムは処理の違いによって大きく二つのルーチンに分けられる。

Matching Routine は入力された独奏者の演奏をその楽譜と matching することにより、独奏者が各音符をどの時刻にどのくらいの音量で演奏したかを認識する。

Planning Routine はマッチングされた音符の演奏時刻と音量によって演奏プランを生成する。

また、独奏者のリハーサルデータは SOLO データとして保存しておく。

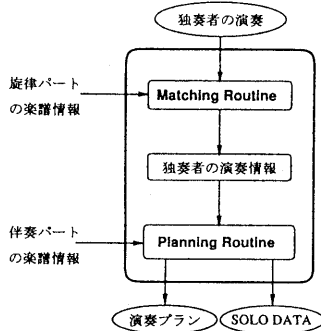


図 1: リハーサルシステムの構成図

3.3.1 演奏プランの時刻計算

	音符番号	1	2	...	n
楽譜データ	演奏時刻	A_1	A_2	...	A_n
	音符間隔	a_1	a_2	...	a_n
リハーサルデータ	演奏時刻	B_1	B_2	...	B_n
	音符間隔	b_1	b_2	...	b_n
	間隔比	r_1	r_2	...	r_n

表 1: 演奏プランの時刻計算

音符番号とは、マッチングした音符を順序付けしたものである。音符間隔は次の音符の演奏時刻との間隔で、

$$a_i = A_{i+1} - A_i$$

$$b_i = B_{i+1} - B_i$$

と表すことができる。間隔比は、楽譜データでの音符間隔とリハーサルデータでの音符間隔との比で、

$$r_i = \frac{b_i}{a_i}$$

である。

$0 \leq k \leq n-1$ において $A_k < T < A_{k+1}$ である時刻 T で楽譜データ中に伴奏パートの音符が存在する場合、この音符の演奏プラン上の演奏時刻 X は

$$X = B_k + (T - A_k) \times r_k$$

のように計算される。

3.3.2 演奏プランの音量計算

音符番号	1	2	...	n
楽譜データ音量	v_1	v_2	...	v_n
リハーサルデータ音量	w_1	w_2	...	w_n
音量比	R_1	R_2	...	R_n

表 2: 演奏プランの音量計算

音量比は楽譜データとリハーサルデータの音量の比

$$R_i = \frac{w_i}{v_i}$$

である。

k 番目にマッチングされた音符と $k+1$ 番目にマッチングされた音符の間の時刻において楽譜データ中に伴奏パートの音符が存在するとする。この音符の楽譜上の音量を p としたとき、演奏プラン上の音量 q は

$$q = p \times R_k$$

のように計算される。

3.3.3 演奏表情の模倣

旋律のある音楽構造が伴奏者によっても演奏される場合がある。その時、実際には伴奏者が旋律を演奏して独奏者は伴奏を行う。すなわち、伴奏パートは自主性を持つ。このシステムでは、伴奏パート中のその音楽構造の演奏表情は独奏者のリハーサル時の演奏表情を真似するようになっている。具体的には、伴奏パートの各音符の長さは対応する旋律パートの音符の長さと同じようにし、伴奏パートの各音符の音量は、対応する旋律音符の音量 / 楽譜上のその旋律音符の音量 \times 楽譜上の伴奏音符の音量のように計算される。

3.3.4 速度変化位置の指定

伴奏パートがどこで旋律のテンポを追従しなければならないかを指定する必要がある。基本的には、旋律パートの各音楽構造の始点を指定する。例外として、伴奏パートに自主性がある部分に対応する旋律部分では指定しない。実際、そのときの伴奏パートは旋律のテンポを追従しなくても良いからである。

3.4 人間とのアンサンブル

独奏者の演奏の入力はシンセサイザーのキーボードや MIDI accordion や MIDI saxphone などを用いて行う。人間の独奏とアンサンブルを行うとき、独奏者の演奏がシステムに入力され、システムはこの演奏に合わせてリアルタイムで伴奏パートの演奏を出力する。ここで独奏者の演奏（入力）と伴奏（出力）は共に MIDI 信号である。

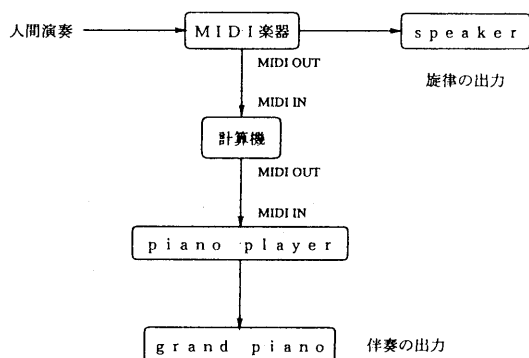


図 2: 自動伴奏システムの構成図

システムは独奏者の演奏を SOLO データとマッチングすることにより、独奏者が各音符をどの時刻に演奏しているのかを認識する。そして、実演奏の 1 拍ごとの長さや SOLO データ上の 1 拍ごとの長さを比較して、独奏者のテンポを検知する。ただし、通常は 1 拍を 1 単位としているが、テンポの遅い曲の場合は楽譜上の 1 拍を 2 分割したり、逆に速い曲の場合には楽譜上の 2 拍をまとめて 1 単位としている。

システムは、速度変化が指定された音楽構造始点からの 1 拍目のテンポをその音楽構造の基礎テンポとして演奏プランを修正し、伴奏を行う。指定されていない部分ではシステムは独奏者のテンポを追従せずに、その前の基礎テンポで演奏プランを修正し伴奏を行う。

独奏者の演奏にミスタッチがあれば、その音は無視されて、速度計算もしない。ただし、伴奏パートの演奏速度を変更可能にするために、指定された音楽構造始点からの 1 拍目にはミスタッチがないという条件を設ける。

ピアノで伴奏する場合には、0.5 秒の遅れを無

くすために、出力すべき時刻より 0.5 秒早めに伴奏パートの音符をピアノに送る。

3.5 考察

実験の結果、このシステムはより自然で人間らしいアンサンブルを実現することができた。

しかしこのシステムは未完成で、今のところ単旋律の楽曲に対する伴奏しか演奏できないことが今後の課題として残された。また、伴奏パート中に独奏者の演奏する音楽構造と類似した構造が少ない曲、つまり旋律パートを真似できる部分が少ない曲の場合には、その伴奏パートの音楽表情が不十分である。人間の伴奏者の演奏を学習する方法を検討したい。

4 自動連弾システム

4.1 開発の背景とシステムの概要

PSYCHE プロジェクトの研究成果である自動伴奏システムはシンセサイザーを対象にしているが、MIDI 対応グランドピアノを演奏装置として用いると演奏の芸術性がさらに高まると思われる。しかしピアノを制御するピアノプレーヤーが、計算機からの制御データを受信してから 0.5 秒後にピアノを制御する構造になっているため、人間の演奏にリアルタイムで反応する自動伴奏システムの実現は不可能である。

そこで、人間の演奏データをリハーサルデータとしてあらかじめ計算機に与えておき、本番の演奏では将来のテンポを予測しつつ人間との連弾を行うシステムを考案した。その結果、曲のフレーズ間の「間」の伸縮の予測が非常に困難であることがわかった。

そこで今度は、「間」の伸縮はテンポの変化に関係があるのではなく、むしろ演奏の「ズレ」であるという新たな観点から、リハーサルと本番とのズレを検出し、可能な限り早くそのズレを補正して連弾を行うシステムを開発した。

このシステムの動作は、本番前のリハーサルと、人間との連弾の二つに大別される。以下、順次説明する。

以後、人間が演奏するパートを PRIMO パー

ト、計算機が演奏するパートを SECONDO パートと呼ぶ。

4.2 リハーサル

このシステムでも前出の自動伴奏システムと同様に、リハーサルの概念を導入している。本番の演奏の前にあらかじめ人間の演奏者に PRIMO パートを演奏してもらい、その演奏データを計算機がリハーサル・データとして取得する。

リハーサル時におけるシステムへのデータ入出力を図 3 に示す。

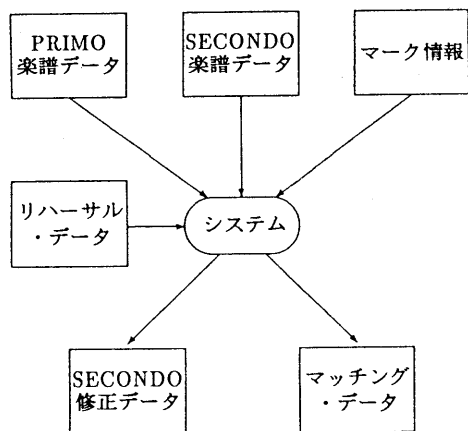


図 3: データ入出力図 (リハーサル時)

4.2.1 マーク情報

PRIMO と SECONDO の楽譜情報以外に、ズレを検出する音符を小節番号と小節内での位置の組で記述した物をマーク情報としてシステムに与える。システムはマーク情報を基に楽譜情報を走査し、ズレ検出点のタイミングと音高とを抽出してリハーサルに備える。

現在のところ、ズレを検出する音符は各小節の 1 拍目とし、マーク情報は人間がテキスト・エディタで記述する方式になっている。

4.2.2 リハーサル・データから得る情報

システムがリハーサル・データから抽出する情報は、各小節の長さ、人間との連弾時にマッチングの目標とするズレ補正点の演奏タイミングである。これをマッチング・データと呼ぶ。

また人間との連弾に備えて、システムはリハーサル・データを参考にして SECONDO の楽譜情報を修正しておく。

4.3 人間との連弾

リハーサルが終わると、人間との連弾を行う。システムへのデータ入出力は図 4 に示す。

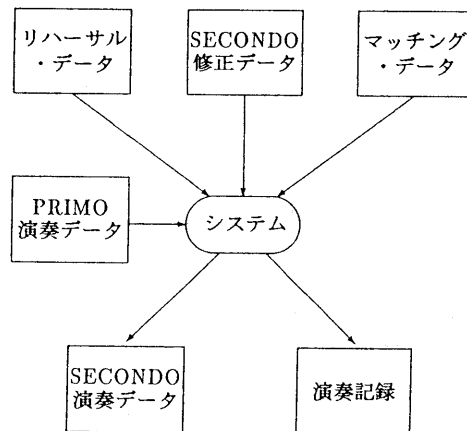


図 4: データ入出力図 (人間との連弾時)

4.3.1 マッチング

連弾の最中、システムは人間の演奏データをモニタして、計算機内に持っているズレ検出音符が弾かれるのを待つ。このマッチングは随時行っているわけではなく、マッチングを行うタイミングと目標とするズレ音符を順次変えていく。これはリハーサル時に得たマッチング・データを参照して決める。

また、マッチングを行う上でマッチング・ウィンドウという時間幅を設けており、この時間幅内に目標の音符が弾かれない場合は、システムは人

間が弾き忘れたと判断し、目標を次のズレ検出音符に移して連弾を続ける。ここで、曲の全体的なテンポやズレ検出音符と同じ音がどれくらい近傍にあるかといったことを考慮できるように、マッチング・ウィンドウの幅は可変になっている。

4.3.2 ズレの補正

今、第 m 番目の小節を演奏しているとする。この小節のズレ検出音符が小節の先頭から n 番目であり、そのタイミングがリハーサルで $t(m, n)$ 、実演で $T(m, n)$ であったとき、ズレは

$$\Delta(m, n) = T(m, n) - t(m, n)$$

と表せる。

すると、この音符以降の音符のタイミングは、

$$T(m, n+k) = t(m, n+k) + \Delta(m, n), \\ (k = 1, 2, \dots)$$

というように、リハーサル時のタイミングに $\Delta(m, n)$ を加算することで予測できる。

このようにして $T(m, n+k)$ が予測できたら、ピアノプレーヤーの遅れの分早くデータを送信すれば良い。例えばピアノプレーヤーの遅れが 0.5 秒ならば、その時刻は

$$T(m, n+k) - 0.5(\text{sec.})$$

となる。補正の結果、データを送信すべき時刻が既に過ぎてしまった場合は、その音符は演奏しない。

4.4 考察

このシステムを構築し、ピアニストの方に使用していただいた。その結果、シンセサイザーを用いた既存の伴奏システムに較べて芸術的な連弾を実現することができた。

しかし、ズレの補正だけでは処理がまだ不完全であり、満足のいく連弾ではなかった。今後は、他の予測方法と組み合わせたシステムを構築していきたい。

また、リハーサルは何度も行えるが、現在のところリハーサル同士は何の影響も及ぼさない。今後は、システムが数種類のリハーサル・データを

参照して人間の演奏の傾向をつかめるように改良していきたい。

また、当プロジェクトでは主にパーソナル・コンピュータを中心にシステムを構成しているが、今回のような自動伴奏の分野では、その処理能力の限界まできていると思われる。今回のシステムでも、その兆候と考えられる事があった。

5 おわりに

以上、PSYCHE プロジェクト内のアンサンブルシステム開発の現状について、特に楽曲構造を重視した研究について述べた。これらの研究はそれぞれ単独で存在するのではなく、互いに処理が類似しており、関連付けられるものである。

今後は、楽曲構造に基づいた演奏表情の抽出をより良く自動演奏へ反映させた、計算機による表情豊かで美しい演奏を目標として実験を進めたい。また、アンサンブルシステムを使用して行った演奏のデータを図形的表現の方法によって視覚的に解析することも考えている。

また、オブジェクト指向の概念を用いて様々な研究を統合した音楽情報処理システムの構築も目指している [1]。

参考文献

- [1] 平賀 瑠美, 五十嵐 滋, 松浦 陽平: 音楽情報処理とオブジェクト指向, 第 37 回プログラミング・シンポジウム報告集, pp. 207-214, 1996.
- [2] 五十嵐 滋, 小川 大典, 戴 岡, 彌富 あかね, 松浦 陽平: 楽曲構造に基づく演奏表情分析と自動演奏への応用, 情報処理学会第 52 回全国大会 (発表予定), 1996.