

マルコフ過程と GA を組み合わせた楽曲再生順序最適化

石井充[†] 松尾和洋[†]金沢工業大学 情報工学科[†]

1. 序論

モバイル音楽プレイヤーが広く使われるようになってきており、既存のテープなどのメディアに比べ、格段に多くの楽曲を持ち運べるようになってきている。ある機種 of モバイル音楽プレイヤーでは、10000 曲以上もの楽曲を保存することができる。

このように多数の楽曲を保存できるようになると、それらの楽曲から個人の嗜好に合わせた楽曲を選び出すのが重要になってくる。これは、単に個人の嗜好にあった曲を選べばそれだけでよいというものではなく、それらの楽曲を個人の嗜好に合わせた順序で再生する必要がある。一般的に、ジャンルが大きく異なる楽曲を連続して再生することはないだろうと考えられるが、それ以上の詳細は個人の嗜好に依存する部分が大きく、ジャンルによる分類では不十分であろう。

こういった問題に対処するために、多くのモバイル音楽プレイヤーでは、あらかじめ好みの順番で楽曲のリストを作成しておく機能が存在する。しかしながら、これは手作業によるものであり、個人がはっきりと認識している固定された順序で楽曲が流れるだけである。

時には順序が変わって意外な楽しみが出てきたり、意外な選曲がされて個人の潜在的な嗜好を見出すという、デジタル音楽プレイヤーの楽しみの一部分が失われてしまう。

そこで、あらかじめ決めた固定された順序に依存するのではなく、自動的にユーザーの嗜好を読み取って楽曲を提示してくれるシステムが作成できないかを検討した。

2. 楽曲選択の基本的な手順

楽曲を選択するためには、再生順序に関するユーザーの嗜好を取得する必要があるが、このために、ユーザーの好みでない順番で楽曲が再生された場合、ユーザーはその楽曲を最後まで聞かずにスキップするであろうということを利用する。たとえば、A, B, C という 3 曲がある場合に

- A が流れ最後まで聞かれた
- 次に B が流れ始めた
- B を最後まで聞かずにスキップした
- 次に C が流れ始めた
- C を最後まで聞いた

とすれば、ユーザーは AB という曲順は好みではないが、AC という曲順は好みであるといえる。

これを定量的に記述するには、A が流れたという条件の下で、B, C が選ばれる確率を定義し、これをユーザーの選択に合わせて最適化して行けばよい。

すなわち、 $P(A|B)$ という確率と、 $P(A|C)$ という確率を考え、A が最後まで聞かれたときに、乱数 r を発生させて、 $r < P(A|B)$ ならば B を選び、 $P(A|B) < r < P(A|B) + P(A|C)$ なら C を選ぶというようにする。

B が選ばれた後に B がスキップされれば、ある割合 δ だけ確率 $P(A|B)$ を下げ、逆に C が選ばれた後に C が最後まで聞かれれば確率 $P(A|C)$ を上げるようにする。

このようにすれば、楽曲を選択していくたびに、 $P(A|B)$ 等が調整されて、ユーザーの嗜好に合致した確率へと収束していくものと期待される。

このように直前の曲のみを参考にして次の曲を決めるのが単純マルコフ過程による楽曲選択である。

3. 単純マルコフ過程の問題点

ユーザーの嗜好する曲順は一般的には完全に確定しているわけではなく、あくまでも確率モデルとしてしか記述できないものと考えられるが、上記のアルゴリズムが現実的に機能するかどうかを調べるために、好みの曲順が完全に確定しているものとして収束判定を行った。

具体的には、さまざまな曲順を乱数で発生させることによって確定させ、初期状態として、 $P(A|A)=P(A|B)=P(A|C)=\dots$ として、再生される曲順が完全に収束するかどうかを調べた。

その結果、どれだけ確率の調整を繰り返しても十分な収束に至らないケースが少なくなかった。

その原因として、直前の曲のみで次の曲が完全に決定されるわけではなく、さらにその前の曲に依存する場合がある事があげられる。

たとえば3曲 A, B, C がある場合に、BACAB という曲順が嗜好されるとする。この場合、2曲目の A の後に C が選ばれ、4曲目の A の後には B が選ばれている。直前の楽曲のみを参考にする単純マルコフ過程では、 $P(A|B)=P(A|C)$ となるので、A の後に B と C が等確率で選択されてしまう。その結果、BABAC や BACAC といった、本来嗜好されている曲順と異なるものがかなりの頻度で現れてしまい、収束を妨げているものと考えられる。

4. GA による補正

この問題を解決するには、直前の曲だけでなく、さらにその前に再生されたいくつかの曲を参考にすればよい。

上の例であれば、BA の後には C だけが選ばれ、CA の後には B だけが選ばれるので、 $P(BA|C)=P(CA|B)=1$, $P(BA|B)=P(CA|C)=0$ となつて、曲順が正しい順序に決定されるはずである。

しかし、N曲が存在するときに、過去の $n-1$ 曲までを考慮に入れると、確率を保存するメモリー領域だけで、 N^n を必要とし、N および n が大きくなると、メモリーの使用に制約が大きいモバイル音楽プレイヤーでは現実的に扱えるサイズでなくなってしまう。

この問題を解決するために、再生頻度の高い楽曲順序についてのみ過去の多くの曲を参照し、他の曲については、直前の曲のみから次に再生する曲を選択することにする。再生頻度の低い楽曲が最後まで聞かれた場合には、次に選択される楽曲が適切でなくなる可能性は依然として残るが、もともと再生頻度が低いので、全体として誤選択を起こす割合に対する寄与は低いものと考えられ、メモリー容量との関係で現実的な妥協策となりえるものと期待される。

具体的には、遺伝的アルゴリズム (GA) を用いてエリート遺伝子を選択する手法により、再生頻度の高い楽曲順序を選び出すことにした。すなわち、

- 長さ5の遺伝子配列を複数用意
- 曲の選択がされるたびに以下の手順を踏む
- 過去の楽曲再生順序と一致するものに高いスコアを与える
- スコアの高い遺伝子を一定割合残す
- それらを組み替えて次世代を作る
- 同じ楽曲順序の遺伝子があればひとつだけを残し、他の遺伝子については既存の遺伝子を再交差させて新しく作成する

を繰り返して遺伝子がより多くの楽曲順序を反映するようにする。

楽曲選択に当たっては、遺伝子と比較して高いスコアを持つ遺伝子に一致する曲順があれば、より高い確率で遺伝子配列上の次の楽曲が選択されるようにする。

その結果、パラメータセットである、遺伝子数・エリート遺伝子の割合・交差確率・突然変異確率を適切に選べば、的確な収束が可能となることがわかった。