

北海道工業大学 ○木下 雄太, 北海道工業大学 畑 大介, 北海道工業大学 木下 正博, 北海道工業大学 北守 一隆

要旨

本研究では音情報からのセンシングによる状況推定を目的としている。物体音源センサは物体の振動(物体音源)から伝搬された波を観測し、シミュレーションを繰り返し、このセンサによる音情報を蓄積し、照合することにより状況を同定するモデルを想定している。

1. 緒言

群ロボットの知覚システムを構成するものとしてセンサがあり視覚、聴覚等の入力機能を提供している。周囲の状況に合わせた制御をおこなうためにはセンサから得られる入力情報を知覚システムに連動させる処理が必要となる。本稿では音情報から状況の推定を行うため、音の種類(What 情報)ではなく音の状態(How 情報)に着目して、物体の振動(Sourced Simulation)と伝搬(Transmitted Simulation)のシミュレーションを行い、物体音源を構成するパラメータを変化させた時に得られる状況音をサンプリングする方法について述べている。

2. シミュレーションによる物体音源の生成

その音がどのような状況で発生しているのか、音の状態(How 情報)を推定するためにはコンピュータ・シミュレーションを使い音物体音源を生成し、パラメータを変化させたときに得られる状況音を蓄積したデータベースから状況の推定を行う。(図.1)推定できる要素は物体が置かれていた位置や落下速度、衝撃の度合いなどがある。物体音源のシミュレーションのことをSS(Sourced Simulation)と呼んでいる。このSSのある時間における音源を $S(t)$ と置き、基本音情報として、正弦波振動としている。この時の物体音源式を $S(t) = A \sin(\omega_0 t)$ と表した。音は1秒間の振動数である周波数によって表現され、周波数は音程を表す振動と振幅が基本要素となっている。 A は音量を表し $A=1$ と仮定し、 ω_0 は音源の角周波数を表す。

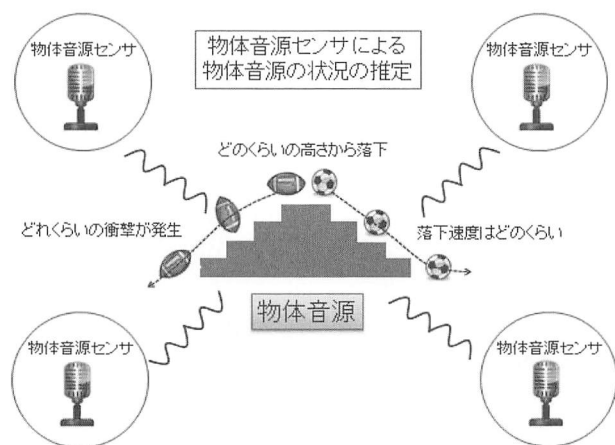


図1. 物体音源の状況の推定

3. 物体音源センサによる音サンプリング

3.1 シミュレーションによる音サンプリング

物体音源センサで行われているサンプリングは一般的にサンプリングと呼ばれているマイクロフォンを使用した録音収集でない。その代わりに物体音源センサで行われるサンプリングは物体音源の情報を抜きだしてwavファイルの形式で書き込みを行う。

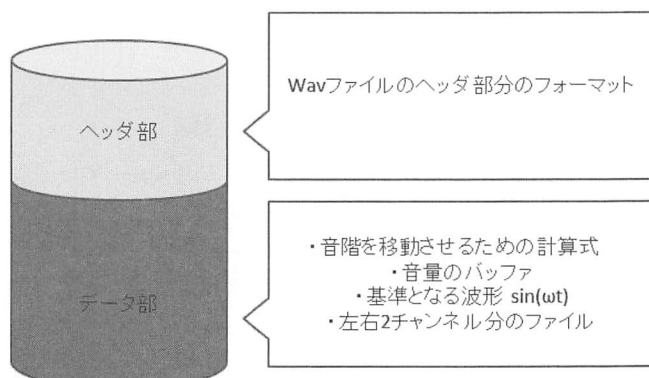
3.2 wavファイルへの直接入力によるサンプリング

物体音源センサはwavファイルに直接、数値を入力することによって音源を生成している。この時のサンプリングは数学的なアプローチによって入力から出力まですべてデジタル上で音源化されることに特徴づけられる。また、今回はscalaというプログラミング言語を使用した。以下に作成したプログラムの要点を記す。

物体音源の条件	実際の数値
サンプリング周波数	441kHz
再生時間	8秒
1秒間に送られるデータ(8秒間)	176400
トータルデータ(8秒間)	1411200

表1. 物体音源の条件

今回の物体音源の条件は表1の項目とする。サンプリング周波数はCDなどでも使われている一番高品質な数値441kHzを採用している。今回は再生時間を8秒と指定している。1秒間に送られる、データはサンプリング周波数に 2×2 と計算された数値使われている。これは左右スピーカー2チャンネル分と上下32768に相当する音量を表す2バイト分の領域が確保されていることを示す。



wavファイル

図2. wavファイルを使った音楽データの構造

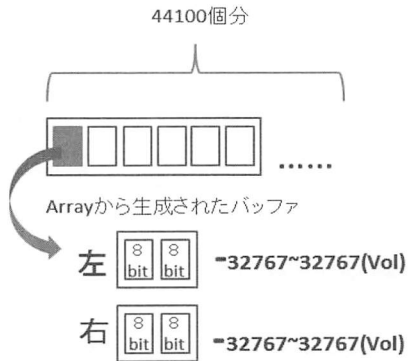
図.2 は今回作成した wav ファイルの概観である。構造は大きく分けて 2 分割できる。wav としての体裁をなすためのヘッダ部分と数値を入力するデータ部である。以下ヘッダ部分の概要とデータ部詳細について以下に記す。

```
case class hD(name:String, v:Array[Byte])

val header = List(
  hD("Chunk ID",    s4("RIFF")),
  hD("Chunk Data Size", b4(bPlay+50)),
  hD("RIFF Type",    s4("WAVE")),
  hD("fmt -Chunk",   s4("fmt ")),
  :
  :
)
```

図 3. 音楽ファイルのヘッダ構造

図.3 はヘッダ部分の一部である。case class hD とは新しく定義された hD というオブジェクトのリストである。hD は String と Byte の 2 つ型からなるオブジェクトであり、これは wav ファイルのフォーマットに準じた形で出力される。この場合は一番上の hD("Chunk ID", s4("RIFF"))では s4 に入力された文字列をバイトにして配列するという機能を持たせているので、RIFF という文字列はバイト形式で出力される。



```
以下のプログラムから
176400毎秒のデータが送られている。
def s4(x:String)=x.map(_>toByte).toArray
これが4秒間行われる。
d(4*i) = (x%256).toByte
d(4*i+1) = (x/256).toByte
d(4*i+2) = (x%256).toByte
d(4*i+3) = (x/256).toByte
```

図 4. 音楽ファイルのデータ構造

図.4 は表.1 の条件を操作するための「機能」についての説明している。これは配列である Array の構造に対応したバッファとなっている。バッファはサンプリング周波数に対応する 44100×4 個の領域を持っている。この中の一つの領域に注目した場合、その構造は左右のチャンネルを表す領域が存在している。またそれら左右を表す領域はそれぞれ 8bit ずつの 2byte の領域を確保している。この計 4 つの 16bit の領域それぞれが音量を表しており、-32768~32767 までの範囲のボリュームが表現できる。今回設定したボリュームは 30000 である。この機能は 1 秒間に 176000 のデータ

ータを wav ファイルへ入力するため、この機能から wav ファイルデータ部が形作られることになる。また d(4*i)から始まるコードでは def s4 の操作が 8 秒間分繰り返されることと、左右のステレオチャンネルを持っていることが記されている。

3. 3 移動する音源の音サンプリング

音源が移動していて、観測者が固定されている場合は直接的なサンプリングだけでは、実際に耳にするような音響効果表現しきれない。このような物理状況から推定を行うために観測者と音源の位置関係を明らかにしなければならない。その位置関係をモデル化して表したのが図.5 である。

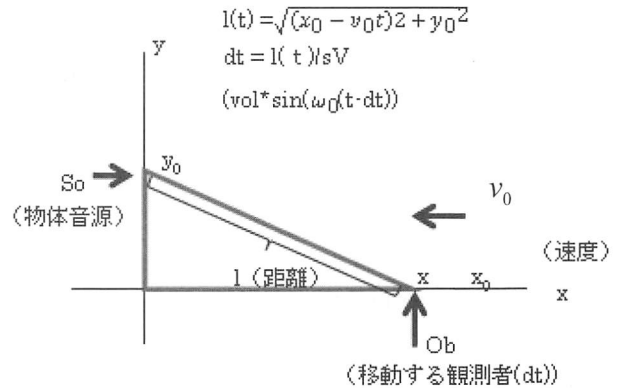


図 5. 音サンプリングと距離

図.5 ではまずピタゴラスの定理から l つまり距離を導き出す。この l に対して音速で割ることにより原音と比べて伝わるまでの時間 dt が導かれる。音源からの時刻からこの dt を引くことによって観測者(Ob)から見ると音源の dt 前の音が届いている。音源が移動している場合は dt は遅れて観測される。このモデルでサンプリングされている。

4. 結言

物体音源センサ内で行われているサンプリングは、一般的なサンプリングとは違い、マイクロフォンによる録音収録ではない。すべてシミュレーションを行って得られるサンプリングである。これらはすべてデジタル上で数値から生成される。このことから実世界では望めない音源、また収録することが難しい音源を入手できる。また、シミュレーション上ではどのような状況も再現できるという性質を利用することによって、一般的な手法のサンプリングで得られた音源と物体音源センサ上で生成された音とを比較することによって状況推定が可能である。

参考文献

Visual Basic で物理がわかる 音波シミュレーション入門
吉澤純夫 著