

4 課題: 簡単な音楽生成プログラム (3)

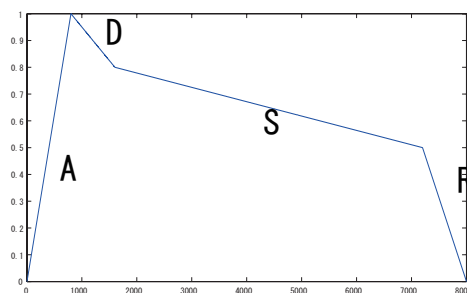
ここまでに作成した楽音生成関数 `note` は、与えられた持続時間の間、ずっと同じ音量で音を生成する。

しかし、実際の楽器音は、発音から時間をおって音量を変化させる。この変化を模擬するのに、ADSR エンベロープと呼ばれる 4 つの区分をもった形がよく用いられる。

このエンベロープをあらわす関数を $E(t)$ とすると、 $E(t)$ にしたがって変化する周波数 262 Hz (ド) の音の振幅は、次の式で計算できる。

$$x(t) = E(t) \cos(2\pi f_0 t + \phi) \quad (1)$$

図の ADSR エンベロープを指定する簡単な関数を考える。



図のエンベロープでは、各区分の相対継続時間にしたがって各区分の境界の頂点を直線で結んでいる。

```
function env = ADSR(format, fs, dur)
% ADSR Function to generate an ADSR envelope
% usage:
%   env = ADSR(format, fs, dur)
%format = [ attack_time attack_level;
%           decay_time decay_level;
%           sustain_time sustain_level;
%           release_time release_level ]
%   時間設定は全体が 1.0 となるような相対時間
%   レベルは 0 から 1.0 の間
%   fs = sampling frequency
%   dur = duration of the waveform in seconds
env = zeros(1, fs*dur);
fin = 1;
amp = 0;
for i = 1:4,
    start = fin;
    range = format(i,1) * dur * fs;
    fin = start + range;
    env(start:fin) = (1)
    amp = format(i,2);
end
```

課題 5 (1) の部分を埋めて関数を完成させよ。

この関数は例えば、

```
fs = 8000;  
dur = 1;  
t = 0:1/fs:dur;  
F = [0.1 1.0; 0.1 0.8; 0.7 0.5; 0.1 0];  
sound(ADSR(F,fs,dur).*sin(2*pi*440*t))
```

というように利用できる。

課題 6 この関数を用いて `note` 関数を拡張せよ。

```
function wave = note(num, dur, fs, ADSR_format)  
(以下略)
```

のように、エンベロープの指定を引数でわたすようにする。