ks

December 9, 2021

1 The Karplus-Strong Algorithm

The Karplus-Strong algorithm is a simple digital feedback loop with an internal buffer of M samples. The buffer is filled with a set of initial values and the loop, when running, produces an arbitraryly long output signal. Although elementary, the K-S loop can be used to synthesize interesting musical sounds as we will see in this notebook.

Let's start with a basic implementation of the K-S loop:

```
In [1]: def KS_1(x, N):
            # given the initial buffer x, produce a N-sample output
            #  by concatenating identical copies of the buffer
            y = x
            while len(y) < N:
                # keep appending until we reach or exceed the required length
                y = np.append(y, x)
            # trim the excess
            y = y[0:N+1]
            return y
```

OK, let's try it out right away! Yet, however impatient we may be, we still need to do a few things. First we need to include the necessary Python libraries:

```
In [2]: %matplotlib inline
        import matplotlib
        import matplotlib.pyplot as plt
        import numpy as np
        import IPython
```

```
In [3]: plt.rcParams["figure.figsize"] = (14,4)
```

Then, since we're playing audio, we need to set the internal "clock" of the system, aka the sampling rate:

```
In [4]: Fs = 16000 # 16 KHz sampling rate
```

With this sampling rate, since the period of the generated signal is equal to the length of the inital buffer, we will be able to compute the fundamental frequency of the resulting sound. For instance, if we init the K-S algorithm with a vector of 50 values, the buffer will fit $16000/50 = 320$