

G B



ology@github

Search this website .

All Possible Fretboard Positions

2013-11-07 BY GENE

tl;dr: The [fretting](#) program on [github](#).



For a while now, I have been burning with a desire to see (and practice) every playable guitar fingering position, in a mathematical (not musical chord based) order, for any arrangement of fingers.

This is a combinatorial problem of enumerating the “variations with repetition” but excluding those that are physically impossible to play.

As a programmer, it is exciting to feel “my brain on logic” and in this case, it has been in implementing the rules to flag “unplayable” positions.

```
use Algorithm::Combinatorics qw(variations_with_repetition);  
...  
# Fret groups per finger.  
my $v = variations_with_repetition($range, $nfinger);
```

Given the above code, here is the main loop, that inspects fret groups and joins

them with the finger symbols:

```
while (my $position = $v->next) {  
    # Flag unplayable fret groups.  
    my @infractions = infraction($position);  
    # Interleave the position with fingers.  
    my @pairs = pairs($fingers, $position);  
    ...  
}
```

First here, we flag unplayable fingerings with two rules:

1. Lower fingers on higher frets and higher fingers on lower frets are not allowed.
2. Frets can't be more than the allowed finger span.

When an unplayable fingering is seen, we skip to the next position (unless we are accumulating all the flags).

```
sub infraction {  
    ...  
    FLAG: for my $current (@position) {  
        ....  
        # Rule 1:  
        if (($seen->{$last} > $current && $last_idx <= $finger_idx) || ($se  
            push @flags, BACKWARD_INFRACTION;  
            last FLAG unless $accumulate;  
        }  
        ...  
        # Rule 2:  
        if ($last_idx != $finger_idx && $fret_span > $sum) {  
            push @flags, SPAN_INFRACTION;  
            last FLAG unless $accumulate;  
        }  
    }  
}
```

In order to facilitate “infraction detection” while allowing one or two fret finger spans, I created a function that uses an “adjacent finger span list.”

```

sub sum_span { # Return the "absolute sum" from a finger span list.
    ...
    # Restrict the list based on the relative sizes of the bounds.
    my @new = @list[ $x > $y ? ($y .. $x - 1) : ($x .. $y - 1) ];
    # Get the sum of the bounded list.
    my $sum = sum(@new);
    # Return the sum of the span, less one for multi-finger spans.
    return @new > 1 ? $sum - 1 : $sum;
}

```

This is the output for three fingers in ascending order (i.e. i=index, m=middle, a=ring), covering the first three frets, plus the open string:

```

$ perl fretting --nfret 3 --nfinger 3 --open
0 0 0
0 0 a1
0 0 a2
0 0 a3
0 m1 0
0 m1 a1
0 m1 a2
0 m2 0
0 m2 a2
0 m2 a3
0 m3 0
0 m3 a3
i1 0 0
i1 0 a1
i1 0 a2
i1 0 a3
i1 m1 0
i1 m1 a1
i1 m1 a2
i1 m2 0
i1 m2 a2
i1 m2 a3
i1 m3 0
i1 m3 a3
i2 0 0
i2 0 a1
i2 0 a2
i2 0 a3
i2 m2 0

```

```
i2 m2 a2
i2 m2 a3
i2 m3 0
i2 m3 a3
i3 0 0
i3 0 a1
i3 0 a2
i3 0 a3
i3 m3 0
i3 m3 a3
```

Here are three fingers, in different order, on four frets with no open strings:

```
$ perl fretting --nfr 4 --nfi 3 --fingers 'i a m'
i1 a1 m1
i1 a2 m1
i1 a2 m2
i1 a3 m2
i1 a3 m3
i2 a2 m1
i2 a2 m2
i2 a3 m2
i2 a3 m3
i2 a4 m3
i2 a4 m4
i3 a3 m2
i3 a3 m3
i3 a4 m3
i3 a4 m4
i4 a4 m3
i4 a4 m4
```

Note the familiar lexicographic enumeration, but without impossible spans (for the average hand) or backward fingerings.

Next up: [Diagrams!](#)

FILED UNDER: MATHEMATICS, MUSIC, SOFTWARE
TAGGED WITH: COMBINATORICS, CONSTRAINTS, GUITAR, PERL

Epistemologist-at-large

^ Top