

to converge (abstract)

choose a range of 16 consecutive integers. then create a set of 16 just intervals/frequency-ratios/dyads such that each integer is used once as a numerator and once as a denominator, or, a set of 8 intervals such that each interval is used once as either a numerator or denominator. the numerator must be greater than the denominator and each interval must be distinct.

at first, play these intervals anywhere within the range of the given instrument. over time, the available range for each interval should converge towards its respective natural octave placement given some (possibly phantom) fundamental. the fundamental should not be predetermined. rather, the performers should collectively choose a narrow range within which the fundamental would occur. the final stage of the piece will be tuning the harmonics to the same fundamental. by the end, one should hear a subset of the harmonic series (determined by the chosen range of integers) of whatever fundamental is converged upon.

there should be between 8 and 16 voices/parts. each voice may be assigned any subset of the intervals such that each interval is uniformly distributed throughout. for example, each part might only consist of one interval, or two so long as each interval is given to two parts, etc.

it is preferred that as many parts as possible are played on acoustic instruments. the remaining parts may be synthesized using sine tones. generally, all tones should have an attack that is shorter than the decay in order to articulate the dyads. each dyad should be sustained for 3 to 15 seconds or the natural decay of the instrument. a performance should last 10 minutes or more.

clear; not loud.

michael winter (july, 2013; los angeles)

\*the first performance of the piece (with 2 lap steels, a viola, and computer) used a total range of two and a half octaves. the narrowed range was a minor 2nd (9/8). the computer randomly chose the fundamental (unknown to the performers) within that range and the performers tuned to the computer output.