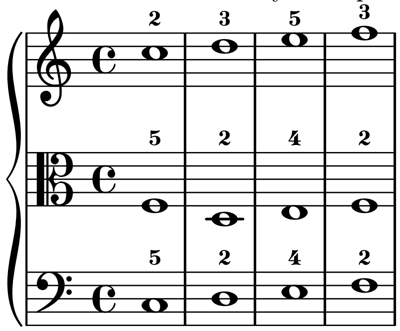
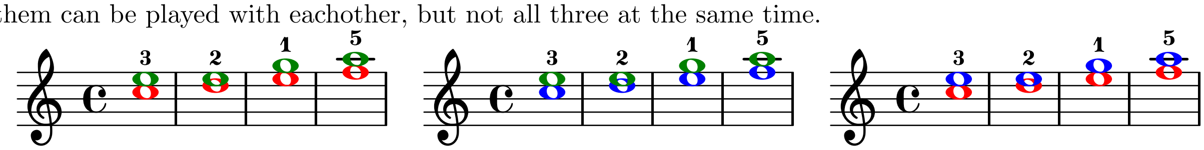
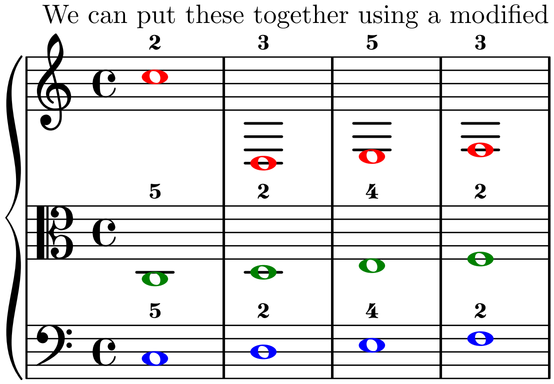
Here's an elementary example of 3-part species counterpoint [1]:



Here are three melody shapes that can be transposed so that any two of them can be played with eachother, but not all three at the same time.

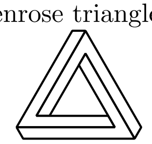


We can put these together using a modified version of the 3-part notation:



Note that the interval numbers no longer add up correctly; the clefs are a fiction to make comparing adjoining staves easier. To work out if the counterpoint is correct, you need to know the intervals and the shape of the melody - you ignore the clefs.

This is an impossible to perform composition in the same way that this penrose triangle is impossible to realize[2]:

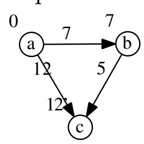


[1] We're ignoring rules about different interval types (diminished fifth), ones that pertain to degree function (leading note stuff), and any ones specific to bass-specific ones. Only rules involving interval and melodic shape are allowed.

[2] cf penrose's paper for the geometrical analogue of what is to follow http://www.iri.upc.edu/people/ros/StructuralTopology/ST17/st17-05-a2-ocr.pdf

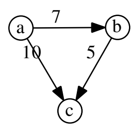
If any two staves are present, the composition can be realized (transposition of one clef might be needed), but all three parts cannot be combined together.

Take the first example given above of a vanilla counterpoint exercise. We can represent the starting pitches with a graph, where vertices are the pitches:

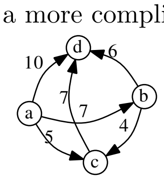


note that if you travel in a loop, adding numbers as you go (or subtracting if you're going against an arrow), you get $7 + 5 - 12 = 0$. It adds up to zero.

For our other graph, we no longer have fixed base positions, so we don't label the vertices, but we can still label the edges, to get this:



Note that the triangle no longer adds to 0, but to 2. Here's a made up graph of a more complicated system:



If we're looking at a graph that shows what intervals are compatible between melodies, and we're looking for triangles where the intervals sum to zero. (For four-part counterpoints, we're looking for tetrahedra where all the constituent triangles sum to zero).

If I have a graph of intervals where all the circular paths sum to zero, I can assign definite pitches to each melody.

Here's the analogy with manifolds and cohomological language:

Manifold	Čech Cohomology	Counterpoint System
atlas		set of melodies
chart	vertex	melody
transition area	edges	harmonizing pair
	nerve	graph of melodies+intervals
	1-cocycle	interval choice
	1-coboundary	compatible intervals
	0-cocycle	assignment of pitches

Define $H(m_1, m_2, i)$ to be true iff m_1 harmonizes with m_2 when at interval i.

It's possible (and fun) to just go and write 3-staff counterpoints by hand, but here's a more algorithmic approach:

- Start with a set M of melodies and a number v which is the maximum voice separation.

- Generate a graph G with a set of vertices M, with labelled edges: $E = \{(m_1 \xrightarrow{i} m_2) | m_1, m_2 \in M, 0 < i \leq v, H(m_1, m_2, i)\}$. You can view this as a directed graph with a 1-cocycle.
- Complete subgraphs of G in which all triangles sum to 0 are realizable counterpoints.

Notes:

- You can leave the diatonic world and take a set of chromatic constraints and that should work pretty nicely as well (it'll avoid weird intervals, at least).
- Also, there's no reason this wouldn't work for higher species.
- There's a balance to be struck between trying to adhere to mathematical formalism and trying to make it clear in and of itself. I don't know how deeply the relationship goes - I don't think cohomology is the natural end-point for it - it branches off from the ech cohomology construction before the cohomology group has been even defined (Penrose's paper basically did as well). I'd probably be more inclined to take a computational direction first, implement something[3] where if you give it a bunch of melody shapes it'll plot a nice graph of them, and see if anything interesting pops up from that. Maybe for certain types of canons certain shapes tend to be produced, for instance.
- You could do a reverse search where you specify a graph and it fills it out with melodies. That could be used to produce some contrapuntal curiosities.

[3] I'm a decent programmer and have implemented lots of counterpoint generation stuff in the past.