Lecture Notes on Pitch-Class Set Theory

Topic 1: Set Classes

John Paul Ito

General Overview

Pitch-class set theory is not well named. It is not a theory about music in any common sense – that is, it is not some set of ideas about music that may or may not be true. It is first and foremost a labeling system. It makes no claims about music itself, but it does make some claims about the basic materials of music, and those claims are objectively, mathematically true, like geometry.

In what follows we first deal with fundamental concepts of pitch-class and interval class, and then we work toward the definition of the set class.

Pitch vs. Pitch Class

We begin with a distinction introduced in the first term of freshman theory, between pitch and pitch class. A **pitch** is a note-name *in a specific register*: for example, F4, the F above middle C. The idea of "an F" is the idea of pitch-class, of note-name leaving aside register. Thus F is a pitch class, F4 is a pitch.

Technically, a **pitch class** is the group (class) of all pitches related by octave equivalence – that is, it is not so much the abstract idea of F's in general, but the group of F's in all audible registers, F0, F1, F2, etc.

For the purposes of pitch-class set theory, we will not distinguish among enharmonically-equivalent spellings of the same pitch or pitch class; F#4 and Gb4 are considered to the same pitch, and F# and Gb are interchangeable names for the same pitch class.

Pitch classes are conveniently represented on a circle as in Example 1; because there are twelve pitch classes, there are twelve positions on the circle, just like a clock. Just as the hands cycle around a clock, going up to 12, on around to 1, and so on, so the an upward chromatic scale goes from G# to A and keeps cycling through the pitch classes.

To avoid unwanted tonal implications, we will often use number pitches. Just as there are two solfege systems for tonal music, using fixed and movable *do*, so there are two possibilities for numbering pitch classes, fixed and movable 0. With fixed 0, C is always 0, and with movable 0, any particularly central pitch in a piece may be called 0. In either case, the pitch class one half-step up is 1 and so forth, so that the pitch class one half-step down is 11. A numbered pitch-class circle is shown in Example 2.

We will most frequently use fixed 0, with C called 0. Memorize the following chart right away.

Pitch-class number	**Pitch-class name**
0	C
1	C#/Db
2	D
3	D#/Eb
4	E
5	F
6	F#/Gb
7	G
8	G#/Ab
9	A
10	A#/Bb
11	B

For convenience, to avoid confusing ’10’ with ‘1’ and ‘0’ in close proximity, we will sometimes abbreviate 10 as ‘t’ and 11 as ‘e’.

Interval Classes

Above I spoke loosely of one pitch class as a half-step above another; if you think carefully about it, this doesn’t really make sense. If pitch-class C is all audible C’s, and pitch-class C# is all audible C#’s, then there is not just one interval between two pitch classes. Instead, there are a rather unmanageable number of intervals present between the pitches in the two pitch-classes: m2 between C4 and C#4, m9 between C4 and C#5, M7 between C#4 and C5, etc. We need a way to talk about intervals between pitch classes, and we will find it by means of the circle of pitch classes.

To visualize two pitch classes, say D and F, find them on the circle; this is shown in Example 3. On the circle, it is easier to see that there are two sensible ways of thinking of the distance between the two; starting at D, we could go clockwise three steps to F, or we could go counterclockwise nine steps to F. (If we start at F, the number of steps is the same, but the three steps are counterclockwise and the nine clockwise.) The relationship between the clock and the circle is especially helpful here because you already know the distance between any two points on the circle.

We define **interval class**, the interval between two pitch classes, as the shorter number of steps around the circle from one pitch class to the other. We abbreviate interval class as 'i.c.', thus, instead of 'major sixth' we will speak of 'i.c. 4'.

As you can see, the largest interval class is 6; if you think you have found a larger interval class than that, you haven't chosen the shorter way around the circle.

Memorize the following chart of interval classes right away.

Interval class	**Interval name**
0	P1 / P8
1	m2 / M7
2	M2 / m7
3	m3 / M6
4	M3 / m6
5	P4 / P5
6	A4/d5

Toward Set Classes

In the remaining portion of these notes, we will be gradually working toward a definition of the set class.

A **pitch-class set** is just that – a set of pitch classes. Unlike a scale, there is no ordering within the set – {A B C} and {C A B} are the same set. A pitch class appears within a set at most once; we may have pitches A3, B3, C4, and B4, but the pitch-class set they represent is {A B C}. What pitch classes to group together as a set is always up to the analyst; a pitch-class set could comprise a harmony, a melody, some mix of harmony and melody, or some other grouping of pitch classes, e.g. the even numbered notes in the second clarinet part.

Pitch-class set theory takes over once the more challenging and creative work has already been done – choosing which pitch classes to include in the set. This is called **segmentation**, and pitch-class set theory tells us nothing about how this is to be done. We will often draw blobs on scores that enclose the sets we are focusing on; these blobs are known informally as **schmoos**.

A **set class** is also just what it says it is – a class, or general grouping, of pitch-class sets that are understood to be functionally equivalent for some purpose. Just as many pitches that are octave-equivalent together comprise one pitch class, so many sets of pitch classes that have other forms of equivalence together comprise one set class. These forms of equivalence will be spelled out below.

Above I emphasized that a set class may represent a melody, a chord, or something else altogether. But it will help to understand why set classes are useful to compare them to chords in tonal music. In tonal music, we have seven scale steps (sometimes modified), each of which is the root of a triad that is diatonically major, minor, or diminished (again subject to alteration), with each of those triads appearing in any of three inversions. This makes for a rich field of possibility, but those possibilities are all broken down into a few manageable categories.

Think now about music in which any group of pitches might form a chord – in 20^{th}-century music, the cluster C4, C#4, D4 can be a chord, while in tonal music it would have (and need) no name. We need, then, a way to sort a very large number of possible chords (really sets, more generally) into a manageable number of categories. Our solution will be to introduce abstractions; these abstractions make different things functionally equivalent.

The first abstraction is to use pitch classes instead of pitches. This helps a great deal. To take three-note chords, on an 88-key keyboard instrument there are 88x87x86=658,416 possible chords with three notes; if we deal only with the twelve pitch-classes, we have 12x11x10=1,320 sets with three different pitch-classes.

Using pitch classes instead of pitches is very helpful, but clearly it isn't sufficient to make a useable category system. The second abstraction is to consider all transpositions to be equivalent. This makes sense in a straightforward way; in tonal music C E G and D F# A are equivalent in a sense (both are major triads), so it makes sense to equate C C# D# and F# G A.

It is relatively easy to see that transposition corresponds to rotation along the circle. Viewed this way, a chord type is a bracket on the circle. We make a bracket that touches C C# and D#, and then we rotate that bracket so that it starts with F#, and we get F# G A. This is shown in Example 4.

This cuts down the number of possibilities by 12 – but that still leaves 110 three-note sets, and three-note sets are among the smaller ones that we'll see.

The final abstraction is to consider inversions to be equivalent. Note here that in pitch-class set theory the word inversion is used differently than in tonal theory. In tonal theory, inversions keep the same pitch classes but shuffle which one appears in the lowest register. In pitch-class set theory, inversions turn chords upside down. Take C E G. To invert this through C is to keep all of the intervals the same but to go in the opposite direction. C is the note we're flipping through, so it stays put; E is a major third above C, so it inverts to Ab a major third below, and G is a perfect fifth above so it inverts to F, a perfect fifth below C. The result is that the major triad C E G inverts to the minor triad F Ab C.

Treating inversions as equivalent is the least intuitive of our abstractions, but even here there are some real aural similarities – major and minor triads are a lot more similar to each other than to most other collections of three random pitch classes. As we use pitch-class set theory you will want to keep an ear out – on a case-by-case basis – for whether or not this abstraction seems justified. As we shall see, it will be on particularly solid ground when analyzing serial music.

Inversion corresponds to flipping the circle around some axis. This is shown in Example 5. Let's stay with our example of C E G inverted around C. The dashed line running through C represents our axis; think of this as a rod running through the paper. Inverting through this axis is equivalent to grabbing the right side of paper and pulling it toward yourself, but with the dashed axis fixed. The paper will rotate until it comes down flat again, but now the side you grabbed and pulled is on the left. This rotation sends our original chord, C E G, indicated by the circles, to C Ab F, indicated by the squares.

Now suppose that we invert C E G through D, shown in Example 6. C is a major second below D, so it goes to E, a major second above. Likewise E goes to C and G goes to A. This gives us E C A, again a minor triad, but a different one.

This is handy – inverting through a different axis is the same as inverting and then transposing, and we already decided that the different transpositions were equivalent. So we can think of a set class as a bracket that we can either rotate around the circle or else flip over and then rotate.

This solves the basic problem – with these three abstractions there are only twelve kinds of three-note set, and though the numbers get larger as the sets get more notes, in general we will be able to deal with the results.

Naming Set Classes

What remains is to find a way of naming the various chords. We'll give them names like 013 and 0268 – we'll take one of the pitch classes and call it 0, and then go around the circle and number the rest of the pitch classes. But again we have a problem with too many possibilities – for C E G we could call any one of the notes 0, and from each one we could number pitch classes going either way around the circle. This gives us six possibilities, shown in Example 7.[1]

We need a way to consistently find just one name for this chord. We'll choose the name that is the most compact. First, we want the last number to be as small as possible. This corresponds to the smallest bracket. To make the last number as small as possible, find the biggest open space between pitch classes on the circle. We'll end up with the smallest final number if we choose one of the pitch classes on the end of the largest open space, and the number pitch-classes going away from that large open space.

---

[1] Why this naming method? Calling one note 0 corresponds to transposing the chord so that it starts on C, and going one way around the circle or the other corresponds to inverting the transposed chord through C – then the numbers in the name are simply the numbers of the pitch classes in the chord.

Let's use C E G as an example again. The biggest open space is from G to C, so we either want to start at C and go clockwise or else start at G and go counterclockwise. This gives us two options: 047 or 037. As you can see from Example 7, these are the two options that give the smallest final number.

At this point we have two options left, and we start going from the left, taking the smaller numbers. This gives us 037 as the name of the chord type.

The procedure for finding the set class name can be summarized as follows:

1) Find the options that make the final number as small as possible

2) Find the options that make the second number as small as possible

3) As needed, continue choosing among options by making the third, fourth, etc. number as small as possible until you are left with only one option; this is the name of the set class.

Examples for Practice Finding Set Classes

{C Eb E Bb}

Let's start with the set {C Eb E Bb}. As shown in Example 8a, the first thing to do is always to plot the pitch classes on the circle.

Figuring out the set-class name is like playing hangman; we have four pitch classes, so we will end up with a set-class name that has four numbers. For now, we'll make a blank for each number, and we'll fill in the first one, which is always, the same, 0.

As our first real step, we make the final number as small as possible, and this means finding the biggest open space on the circle; this is between E and Bb. So we know that either we make E 0 and go counter-clockwise or else we make Bb 0 and go clockwise; either way we can fill in the last blank: 0 _ _ 6. This is shown in Example 8b.

In step 2, we make the second number as small as possible. There are two ways to approach this; we could write out the full set-class name for each of our remaining options, or we could just look at the circle; if we start with Bb and go clockwise, our second number will be 2, but if we start with E and go counterclockwise the second number will be 1, which is smaller; so we fill in 0 1 _ 6, and because we only have one option left, we can fill in the rest: 0146. This is shown in Example 8c.

{D F# Bb}

Our next example is the familiar augmented triad; at this point you should be charting the sets on the circle yourself. As we know, this is a symmetrical chord; each note is a major third away from the others. (Note, though, that the "major third" from F# to Bb is spelled as a diminished fourth; problems like this always arise with traditional spellings of symmetrical chords. Using numbers, 2, 6 and 10 are all separated by i.c. 4's.)

This means that there is no largest space; no matter which number we make 0, and no matter which way we go around the circle, we get the same set-class name: 048.

{C# D E G Bb B}

At this point the steps should be becoming more familiar. We can see right away that the biggest spaces are between E and G and between G and Bb. We can rule out G as a starting point, because no matter which way we go from G we begin with one of our largest spaces – this won't make the second number as small as possible. So our options are going clockwise from Bb or counterclockwise from E, and again we can see right away that we'll start out 01 from Bb but 02 from E. So our set class name starts with Bb and goes clockwise: 013469.

{C D F# A B}

Before reading further, try this one on your own.

This example has been constructed to remind you of the order of the steps. If you go too quickly, you may jump to going for the smallest second number, in which case this would be 0 1 _ _ _ – you either start with B and go clockwise or start with C and go counter-clockwise. This would make either 0137t (starting from B, remembering that 't' abbreviates 10) or else 0136t (starting from C). We'd choose the second one because the fourth number is smaller.

But making the second number as small as possible is the second step, not the first – the first step is to make the last number as small as possible, and the biggest space is between D and F# – our set-class name is 0 _ _ _ 8. We can quickly see that if we start with F# and go counterclockwise we'll get 03…, but if we start at D and go counter-clockwise we'll get 02…, so we should start with D; the set-class name is 02358.

Notation and Terms for Set Classes

So far we've been notating set-class names as strings of numbers, such as 02358. Especially in older literature, you may see these enclosed in brackets, as [02358]. This notation for set class names is also called **interval normal form**, or simply **normal form**, and set classes are also referred to as T_n/T_nI types – this terminology will be explained when we discuss transposition and inversion at greater length in subsequent topics.

There is also a second way of naming set classes, by giving them **Forte numbers**, named for the music theorist Allen Forte. Forte numbers take the form n-m, where n is the size (or **cardinality**) of the set, and m is its "place in line" when the sets are all lined up from the most compact to the least compact. Thus 3-1 is the most compact set of three pitch classes, which we would call 012. The problem with Forte numbers is that unless you have them all memorized, you need to consult a table to find out what the set is.

Until this point we have been using cumbersome descriptions to indicate cardinality of sets; from here on out we will use the following terms, which you should memorize.

Cardinality	**Term**
2	dyad
3	trichord
4	tetrachord
5	pentachord
6	hexachord

Interval Vectors

One of the most important things to know about a set class is what intervals there are between its various pairs of pitch classes. For example, in 012, which we will represent with {C C# D}, there are two i.c. 1's (between C and C# and between C# and D) and one i.c. 2 (between C and D).

We represent the count of intervals in a set class with an **interval vector**, which takes the form <l, m, n, o, p, q>, or simply <l m n o p q>, where l is the number of i.c. 1's, m is the number number of i.c. 2's, and so forth, up to q as the number of i.c. 6's.

The easiest way to compute an interval vector is to use the circle. Let's use the last of our examples of set-class names, the 02358, which is already shown in Example 11.

Once we've plotted the set, we make another hangman-style chart, this one to record tick-marks for each of the interval types:

1 2 3 4 5 6

___ ___ ___ ___ ___ ___

To compute the interval vector, start with one pitch class and go clockwise around the circle finding the interval classes with all of the other pitch classes. We'll start with C; there's an i.c. 2 with D, an i.c. 6 with F#, an i.c. 3 with A, and an i.c. 1 with B. So at the end of our first pass our chart looks like this:

1 2 3 4 5 6

|___ |___ |___ ___ ___ |___

Now we go on to the next pitch class, and again we go clockwise around the circle finding interval classes. <u>The key here is to skip the pitch-classes already used as starting points</u>. We already counted the interval from C to B, so if we count the interval from B to C we'll be double-counting that interval.

On our second pass, going from D, we have an i.c. 4 with F#, an i.c. 5 with A, and an i.c. 3 with B. Now our chart looks like this:

1 2 3 4 5 6

|___ |___ ||___ |___ |___ |___

The third pass starts with F#; we have an i.c. 3 with A ad an i.c. 5 with B.

1 2 3 4 5 6

|__ |__ |||__ |__ ||__ |__

Finally we count the i.c. 2 from A to B and our chart is complete:

1 2 3 4 5 6

|__ ||__ |||__ |__ ||__ |__

We count the tick marks, and we get our interval vector: <1 2 3 1 2 1>.

The Z-Relation

Sometimes two different set classes share an interval vector. When this happens, we say that the sets are **Z-related**. When using Forte numbers, Z-related sets are indicated with a ‘Z’ preceding the second number, for example 4-Z15 (just as you need a table to know what the set class is, you need a table to know which other set class has the same interval vector.) The Z-relation is not flagged notationally when using our standard names for set classes.

As an exercise, verify that 0146 and 0137 are Z-related.