a journal of the Society for Music Theory

Volume 0, Number 10, September 1994

Modifying Interval-Class Vectors of Large Collections to Reflect Registral Proximity Among Pitches

Brian Robison

KEYWORDS: harmony, set theory

ABSTRACT: The twelve-tone operations of transposition and inversion reduce all 12-note collections to one set-class, all 11-note collections to one set-class, and all ten-note collections to one of six set-classes. Yet throughout this century, composers have organized large collections in different ways to produce distinctly characteristic intervallic content. I propose here a modified version of Allen Forte's interval-class vector to preserve some indication of registral proximity among pitches, assigning fractional counts to intervals greater than six semitones in order to derive interval-class vectors which distinguish different sets of a given set-class.

---

[1] The following work-in-progress attempts to bridge an analytic gap between order and disorder: between symmetric pitch matrices or interval cycles on the one hand, and asymmetric, all-interval structures (often labeled as "ad hoc") on the other. Although this discussion treats only large collections, the technique proposed is sufficiently general to apply to sets of any cardinality. The calculation of modified interval-class (ic) vectors for individual pitch sets (as opposed to pitch-class sets) requires computer implementation; Appendix 2 lists the structured BASIC program used to generate the ic vectors given in Appendix 1.

Sets and Set-Classes

[2] Pitch-class (pc) set theory provides powerful abstractive tools for relating pitch collections by operations such as transposition and inversion. However, these same operations reduce all 12-note collections to one set-class, all 11-note collections to one set-class, and all ten-note collections to one of six set-classes distinguished by the interval-class of the excluded pc pair.(1)Especially when expssed as chords in close or relatively even spacing, such large collections often do not lend themselves to perceptually clear segmentation into subsets with more distinctive ic vectors.

[3] Throughout this century, composers have organized large collections in different ways to produce distinctly characteristic intervallic content. These range from the consistent use of one or two ic's between vertically adjacent chord-tones, especially in vertically symmetrical arrangement, as in the music of the late Witold Lutoslawski(2) Olivier Messiaen(3), and others(4), to "all-interval" constructions, typically asymmetrical, as is often the case in verticalizations of 12-tone rows.

[4] pviously proposed functions which distinguish different members of a given set-class, such as Chapman's "above-bass" ic

vector[5] and Morris's INT(n) function[6], provide detailed information about relationships among pitches; however, as set cardinality increases, this detail proliferates geometrically to become unwieldy in practice.

A Geometric Analogy

[5] Many of the difficulties described above result from the conventional mapping of pitch space to pitch-class space, by which one completely collapses the helix of the former onto the circle of the latter. However, for most objects of dimension D, one can choose among multiple projections to a space of dimension D-1. For example, a cube can project onto a plane as any of the following:

1. a single square,
2. two intersecting squares, with corresponding sides parallel and corresponding vertices connected, or
3. a regular hexagon with opposing vertices connected.[7]

[6] Each repsentation has its merits and limitations. The first pserves right angles and equal lengths of some edges, but effectively obscures the original's three-dimensional character. The second pserves all parallelisms and most angles and lengths, but distorts others. The third pserves all parallelisms and repsents all edges as equal in length, but distorts all angles (some become 60 degrees, others 120 degrees).

[7] Returning to the helical model of the pitch continuum, there are likewise several possible approaches. Viewed from the side, the helix appears as a sinusoid. This sinusoidal projection pserves registral position but distorts relative distances (e.g. some semitones appear longer than others). As mentioned pviously, the circular projection pserves interval-class relationships but eliminates all information regarding registral position.

Interval-Class Vectors with Fractional Values

[8] I propose here a modified version of Allen Forte's ic vector[8] which provides some indication of registral proximity among pitches; in the geometric analogy above, it corresponds roughly to viewing the helix from above, but with some element of depth perception. By assigning fractional counts to intervals greater than six semitones, we can derive ic vectors which distinguish different voicings of any given pc collection.[9]

[9] Within each ic, the smallest interval receives a value of one; successively larger intervals are counted at successively smaller values, as shown below for three decrement schemes: one linear, one moderately exponential, and one strongly exponential. The linear scheme uses a steady decrement of .05, which maintains non-zero values for intervals within a limit of ten octaves (corresponding to the conventional range of human hearing). The exponential schemes are based on reciprocals of e[10]raised to 1/5 and 1/2 the index of transposition (see table below). In all of these, the conventional ic vector is taken as expssing dissonant maxima[11], which the composer can emphasize or attenuate by means of registral disposition.[12]

Fractional interval counts

Interval in semitones	Index of transpos.	Decrement schemes: (a) Linear	(b) Moderately exp.	(c) Strongly exp.
1 - 6	0	1.0	1.0	1.0
7 - 12	1	.95	.819	.607
13 - 18	2	.9	.670	.368
19 - 24	3	.85	.549	.223
25 - 30	4	.8	.449	.135
31 - 36	5	.75	.368	.082
37 - 42	6	.7	.301	.050
43 - 48	7	.65	.247	.030

etc.	etc.	etc.	etc.	etc.

Variation Sets

[10] **Example 1** provides ten different voicings of 8-24, displaying varying degrees of symmetry, varying numbers of ic's as adjacencies, and varying ic's emphasized as adjacencies. The first three are Messiaen's *accord de resonance* and two of its inversions.(13) 1d and 1e emphasize ic's 1 and 2; 1f emphasizes perfect fourths, while 1g manifests augmented triads in close spacing. 1h, 1i, and 1j psent several all-interval constructions. For 1a and 1c, the fractional ic vectors (Appendix 1: Tables 1a, 1b, 1c) reflect the tertial structure's relative emphasis by proximity of ic3 over ic1.(14) Like conventional vectors, they fail to distinguish 1a from 1c. However, unlike conventional vectors, and unlike Chapman's and Morris's functions, they do distinguish between 1d and 1e, which contain identical vertical sequences of pitch-classes.

[11] The exponential schemes psent more dramatic distinctions than linear ones, as the fractional ic vectors (Appendix 1: Tables 2a, 2b, 2c) for the 12-note sonorities in **Example 2** illustrate. Again, the voicings vary in regard to symmetry and restriction of ic's as adjacencies. Note that the greater decrements for highly compound intervals produce a wider range of possible values in any given position in the vector. For example, compare the ic6 values in the vectors for the first five sonorities alone (vertically symmetrical 12-note chords with only ic3 and ic5 as adjacencies, spanning comparable registers). The linear scheme produces values between 5 and 6, the moderately exponential scheme produces values between 3.28 and 6, and the strongly exponential scheme produces values between 1.57 and 6. Fractional vectors derived with strongly exponential decrements thus approach a vector calculated by counting adjacencies alone.

Preliminary Implications for Analysis

[12] To ensure meaningful comparison, it's important that an analyst apply either the same scheme or closely comparable ones to all collections in a given context. Conservative linear decrement schemes distinguish different voicings of a given pc-set, but their vectors are still closer to each other than they are to those of other pc-sets, and even to those of voicings of other sets which display highly similar adjacency structures. In contrast, the exponential schemes produce vectors in which the intervallic cardinality (i.e. the sum of the terms of the ic vector) of a chord distributed over a wide range approximates that of a much smaller conventional collection. In extreme cases, pitch sets which belong to different pc-set-classes but which share certain adjacency structures could generate vectors which are closer to each other than to those of other members of their own pc-set-classes, as measured by numerical methods such as Isaacson's IcVSIM function.(15)

[13] However, I do not psume to suggest any standard scheme, since the pferred size and nature of intervallic decrements will vary among individuals and among repertoires. To return to the geometric analogy, these variations correspond to differences among individual perceptions of the relative spacing of the helix's coils and the resulting degree to which aural "perspective" attenuates the dissonance of highly compound intervals. In addition, aspects of timbre and orchestration often motivate different perceptual distances; consider the difference in relative dissonance of any one of the sonorities above as performed on a harpsichord, clavichord, piano, or organ, or as played by bowed strings, muted, *ppp*, or brass choir, *fff*. As with any other feature of analysis, one should note explicitly the information which one chooses to sacrifice for the sake of brevity and clarity.

Appendix 1: Tables

Table 1a

Fractional interval-class vectors of the 10 voicings of 8–24 in Example 1 (linear decrements)

	4	6	4	7	4	3
a	3.6	5.65	3.95	6.6	3.75	2.9
b	3.7	5.6	4	6.5	3.7	3
c	3.6	5.65	3.95	6.6	3.75	2.9

d	4	5.95	4	6.9	3.9	3
e	3.8	5.35	3.7	5.8	3.2	2.5
f	3.8	5.1	3.3	6.0	4	3
g	3.5	5.45	3.5	6.9	3.5	2.8
h	3.7	5.65	3.8	6.3	3.6	2.7
i	3.35	5.05	3.6	6.2	3.8	2.5
j	3.75	5.4	3.55	6.3	3.75	2.9

================================

Table 1b

Fractional interval-class vectors of the 10 voicings of 8-24 in Example 1 (moderately exponential decrements)

	4	6	4	7	4	3
a	2.71	4.80	3.82	5.71	3.19	2.67
b	2.98	4.71	4	5.29	3.10	3
c	2.71	4.80	3.82	5.71	3.19	2.67
d	4	5.82	4	6.64	3.64	3
e	3.34	4.26	3.01	3.75	1.86	1.57
f	3.27	3.49	2.00	4.09	4	3
g	2.44	4.20	2.54	6.64	2.59	2.34
h	3.04	4.80	3.37	4.85	2.86	2.12
i	2.24	3.49	2.86	4.67	3.34	1.75
j	3.16	4.20	2.67	4.85	3.19	2.67

=====================

Table 1c

Fractional interval-class vectors of the 10 voicings of 8–24 in Example 1 (strongly exponential decrements)

	4	6	4	7	4	3
a	1.57	3.56	3.61	4.57	2.44	2.37
b	1.95	3.57	4	3.53	2.45	3
c	1.57	3.56	3.61	4.57	2.44	2.37
d	4	5.61	4	6.21	3.21	3

e	2.74	3.13	2.10	1.72	.67	.64
f	2.43	1.84	.72	2.00	4	3
g	1.18	2.54	1.48	6.21	1.67	1.74
h	2.20	3.56	2.83	3.16	2.06	1.50
i	1.16	2.10	2.06	3.02	2.74	1.19
j	2.34	2.79	1.73	3.16	2.44	2.37

==============================

Table 2a

Fractional interval-class vectors of the 15 voicings of 12-1 provided in Example 2 (linear decrements)

	12	12	12	12	12	6
a	10.35	9.7	10.55	11.2	10.35	5.0
b	10.3	10.5	11.05	10.4	10.8	5.4
c	10.45	9.7	10.85	11.2	10.55	5.0
d	10.65	10.1	11.05	11.2	10.75	5.2
e	10.5	10.4	11.85	10.4	10.5	6
f	10.55	10.1	10.75	11.2	11.05	5.2
g	10.4	10.55	10.85	10.5	11.05	5.5
h	10.75	10.1	10.65	10	9.45	4.8
i	10.15	9.7	10.75	11	10.45	5
j	10.5	10.6	11.85	10.6	10.5	6
k	10.55	10.7	11.1	10.8	11.05	5.3
l	10.95	10.85	10.9	10.6	11	5.6
m	9.10	8.90	9.15	9.15	8.9	4.4
n	10.5	10.65	11.25	10.6	10.7	5.7
o	10.65	10.6	11.1	10.55	10.5	5.4

==============================

Table 2b

Fractional interval-class vectors of the 15 voicings of 12-1 provided in Example 2 (moderately exponential decrements)

	12	12	12	12	12	6
a	7.30	5.80	8.02	9.23	9.00	3.28
b	7.48	7.41	9.26	7.59	8.40	4.24
c	7.53	5.80	8.99	9.23	8.00	3.28
d	7.97	6.58	9.45	9.23	8.39	3.80
e	7.66	7.20	11.46	7.27	7.67	6
f	7.77	6.58	8.45	9.23	9.36	3.80
g	7.26	7.77	9.00	7.65	8.96	4.57
h	8.47	6.58	8.45	7.15	5.66	2.70
i	6.79	5.92	8.79	8.88	7.53	3.50
j	7.38	7.81	11.46	7.27	7.67	6
k	7.57	8.21	9.26	8.41	9.55	3.80
l	8.77	8.60	8.85	7.83	9.07	4.79
m	7.97	8.00	8.45	7.70	7.30	3.80
n	7.75	7.81	9.86	7.92	8.35	5.12
o	7.88	7.96	9.53	7.76	7.84	4.42

==============================

Table 2c

Fractional interval-class vectors of the 15 voicings of 12-1 provided in Example 2 (strongly exponential decrements)

	12	12	12	12	12	6
a	3.95	2.35	5.58	6.32	7.52	1.57
b	4.65	3.85	7.40	4.64	5.74	3.01
c	4.30	2.35	7.26	6.32	5.62	1.57
d	4.86	2.91	8.04	6.32	5.90	2.54
e	4.54	3.55	10.82	3.86	4.68	6
f	4.73	2.91	6.15	6.32	7.58	2.54
g	3.76	4.63	7.36	4.64	6.19	3.64

h	5.75	2.91	6.65	4.47	2.55	.81
i	3.20	2.53	7.08	6.36	4.28	2.37
j	3.69	4.63	10.82	4.82	4.09	6
k	4.08	5.51	7.15	5.48	8.22	1.97
l	6.12	6.04	6.77	4.62	6.91	3.87
m	4.86	5.02	6.40	4.78	4.74	2.54
n	4.78	4.47	8.41	4.87	5.94	4.50
o	4.62	5.06	7.94	4.73	5.17	3.55

Appendix 2: The FRACTION Program

```
DECLARE SUB EnterPitches ()
DECLARE SUB FractionalCounts ()
DECLARE SUB ShowVector ()

DIM SHARED Quant
DIM SHARED IntClass

DIM SHARED PitchArray(3, 30)
  'Thirty is an arbitrary value, selected because
  '  BASIC does not allow matrices of variable dimension.
  'If analyzing collections of more than 30 pitches,
  '  substitute a larger value in this dimension statement.

DIM SHARED IcMatrix(12, 6)

DEF fnPitchClassOrder (p, q)


  'This function enables the FractionalCount routine
  '  to assign fractional counts to their proper positions
  '  in the interval-class matrix.


  IF IntClass = 6 THEN
    SELECT CASE p
      CASE IS < q
        fnPitchClassOrder = p
      CASE IS > q
        fnPitchClassOrder = q
      END SELECT
    EXIT DEF
  END IF
  SELECT CASE p
    CASE IS = (q + IntClass) MOD 12
      fnPitchClassOrder = q
    CASE ELSE
```

```
      fnPitchClassOrder = p
  END SELECT
END DEF


'Program Fraction

  'This program calculates interval-class vectors
  '  using fractional interval-class counts
  '  to reflect emphasis by registral proximity.


  DO
    CLEAR
    CALL EnterPitches
    CALL FractionalCounts
    CALL ShowVector
    PRINT "      "
    INPUT "Process another collection"; Response$
      IF Response$ = "n" THEN EXIT DO
  LOOP
END

SUB EnterPitches
  'This routine accepts pitch data as ordered pairs
  '  of pitch-class (in integer notation)
  '  and register (according to the ASA scheme).
  'It then converts each pair to a position
  '  for calculating intervals.
  CLS
  INPUT "Enter the number of pitches in the collection: "; Quant
  PRINT "Enter each pitch as an ordered pair (pitch-class, register)"
  FOR i = 1 TO Quant
    PRINT "  Pitch No."; i;
    INPUT PitchArray(1, i), PitchArray(2, i)
    PitchArray(3, i) = 12 * PitchArray(2, i) + PitchArray(1, i)
  NEXT i
END SUB

SUB FractionalCounts


  'This routine calculates intervals and assigns their
  '  fractional counts to an interval-class matrix.

  FOR v = 1 TO Quant - 1
    FOR w = v + 1 TO Quant
      Interval = ABS(PitchArray(3, v) - PitchArray(3, w))
      IntClass = Interval MOD 12
      Index = 2 * ((Interval - IntClass) / 12)
        IF IntClass > 6 THEN
          IntClass = 12 - IntClass
          Index = Index + 1
```

```
        END IF
      Delta = Index * .05
      Fraction = 1 - Delta
  'The above two lines implement the linear decrement
  '  scheme discussed in the article text.
  'For moderately exponential decrements, replace them
  '  with the single line

  '   Fraction = 1/(2.718281828# ^ (Index/5))

  'For strongly exponential decrements, replace them
  '  with the single line

  '   Fraction = 1/(2.718281828# ^ (Index/2))

      r = fnPitchClassOrder(PitchArray(1, v), PitchArray(1, w))
        IF Fraction > IcMatrix(r + 1, IntClass) THEN
          IcMatrix(r + 1, IntClass) = Fraction
        END IF
    NEXT w
  NEXT v
END SUB

SUB ShowVector

  'This routine sums the fractional counts for each
  '  interval-class, producing the fractional IcVector.


  PRINT "Fractional IcVector = ";
  DIM sum(6)
  FOR m = 1 TO 6
    sum(m) = 0
    FOR n = 1 TO 12
      sum(m) = sum(m) + IcMatrix(n, m)
    NEXT n
    PRINT sum(m);
  NEXT m
END SUB
```

Brian Robison
Music Department
Lincoln Hall
Cornell University
Ithaca, New York 14853
bcr2@cornell.edu

Footnotes

1. Specifically, these "saturated" set-classes are:

12–1 [12 12 12 12 12 6]

11–1 [10 10 10 10 10 5]

10–1 [9 8 8 8 8 4]

10–2 [8 9 8 8 8 4]

10–3 [8 8 9 8 8 4]

10–4 [8 8 8 9 8 4]

10–5 [8 8 8 8 9 4]

10–6 [8 8 8 8 8 5]

Note that, among set-classes of cardinality 10, the vectors show a *higher* value for the "excluded" pc pair (i.e. by reflexivity, the excluded pair removes only one instance of its own ic, but two instances of every other ic).
Return to text

2. Steven Stucky, *Lutoslawski and his music* (Cambridge: Cambridge University Press, 1981), 114–119.
Return to text

3. Some of these collections arise from interval cycles (as in the chord of alternating fourths and tritones which repeatedly appears in the first movement of his *Turangalila-symphonie*), others from the inherent symmetries of his modes of limited transposition (Messiaen: *Technique de mon langage musical*, Paris: A. Leduc, 1944.).
Return to text

4. Two prominent early collections of limited adjacencies are the 12-note chords that accompany the deaths of the title characters in Alban Berg's *Wozzeck* (ic's 3 and 4, symmetric) and *Lulu* (ic's 1, 5, and 6, asymmetric).
Return to text

5. Chapman, Alan. "Some intervallic aspects of pitch- class set relations." *Journal of music theory* 25 (1981): 275–290.
Return to text

6. Morris, Robert. *Composition with pitch classes: A theory of compositional design*. New Haven: Yale University Press, 1987.
Return to text

7. For the sake of simplicity, these examples assume parallel perspective in projection (i.e. no vanishing point!). Of course, the introduction of one or more vanishing points further multiplies the variety of possible projections.
Return to text

8. Because the system outlined by Forte in *The structure of atonal music* (New Haven: Yale University Press, 1973) has become a lingua franca among theorists, it provides a valuable framework in which to build. Thus, the counting schemes outlined below all produce ic vectors of six values ranging from [0 0 0 0 0 0] for the empty set-class to [12 12 12 12 12 6] for the complete aggregate with all 66 possible intervals occurring in their most compact forms (due to duplication of pitch-classes).
Return to text

9. Although these fractional counts resemble membership functions of fuzzy sets, as set forth by Lotfi Zadeh in his article "Fuzzy sets," *Information and control* 8 (1965), pp. 338–353, I do not mean them to imply a reduced degree of ic membership for compound intervals! Rather, they denote degrees of membership in the fuzzy set of *close intervals*, summed by ic as six scalar cardinalities to produce an ic vector. By extension, the fractional vectors do not imply reduced set-class membership for sets widely dispersed in register, but only lesser membership in the fuzzy set of *closely-spaced sonorities*. For a lucid review of basic concepts of traditional ("crisp") set theory and fuzzy set theory, see George J. Klir and Tina A. Folger: *Fuzzy sets, uncertainty, and information* (Englewood Cliffs: Prentice-Hall, 1988), 1–21.
Return to text

10. *e* is the base of the system of natural logarithms, approximately 2.718 in value. The choice of *e* as a basis for these exponential schemes is purely arbitrary. Empirical research in the perceived distances of compound intervals may suggest alternate decrement functions.
Return to text

11. Here "dissonant" refers loosely to the acoustical phenomenon of critical bandwidth, corresponding to the colloquial notion that a minor second is "crunchier" than a major seventh, and so on.
Return to text

12. Thus, in order to preserve the bounds described in note (7) above, in collections containing more than one pitch of a given pc, the algorithm implemented in the *Fraction* program (Appendix 2) always chooses the more compact form of a given pc pair, e.g. for the collection {C4,D4,B4,C5} under the linear scheme above, the ic vector is [1 1 .95 0 0 0].
Return to text

13. Messiaen, *Technique de mon langage musical*, Vol. 2, Exx. 208, 209, 210. I've transposed Messiaen's Ex. 209 so that it shares the same pitch classes with the collections of Ex. 1a and 1d–1j.
Return to text

14. That is, these two sonorities each (enharmonically) contain fairly compact versions of ic3 (three minor thirds and one major sixth), vs. compound versions of ic1 (no minor seconds, two major sevenths, one minor ninth, and one major fourteenth).
Return to text

15. Isaacson, Eric. "Similarity of interval-class content between pitch-class sets: The IcVSIM function." *Journal of music theory* 34 (1990), 1–28.
Return to text

Prepared by Cynthia Gonzales and Rebecca Flore, Editorial Assistants