Aspects of Post-Tonal Canon Systems

Robert D. Morris

The study of canon is an important part of our understanding of the structure of post-tonal music, for the serial transforms of P, I, R, RI, and rotation have their roots in the concept of canon, especially when they are used in contrapuntal passages in works by Webern, Schoenberg, Berg, Stravinsky, Wolpe, and Babbitt. Among recent studies of canons is my paper "The Structure of First-Species Canon in Modal, Tonal and Atonal Musics," published in volume 9 of this journal, which provided algorithms for generating two-voice, note against note, transpositional canons in various styles. In this paper I take this work further and propose methods to construct two-voice canons by inversion and transpositional canons in any number of voices and in hybrid species.

Transpositional Canon Systems

Definition 1. A transpositional pc canon system is denoted by the function TCS(t, V), where t is the interval of transposition and V is the set of vertical intervals permitted between simultaneously sounding voices.

The definition tells us that a *transpositional pc canon system* is denoted by the function TCS(t, V), and it generates note-against-note canons according to a *transpositional canon graph* determined by t and V.[1, 2] Example 1a shows the graph of a canon system where t = A, and V = the set of intervals{1,6,B}. One constructs a canonic subject by following the arrows on the graph; this yields a

[1] In Morris 1997, V is called S, and is actually defined as the set of interval-classes between the voices—a more limited definition since, if $i \in V$, then $-i \in V$.

[2] In the sequel, I shall just refer to canon systems and canon graphs without mentioning the word "transpositional" to save time and space. When we discuss inversional canon systems, I will call them "i-canon systems."

series of intervals that are the adjacent intervals of the subject. Example 1b shows a subject and a canon that satisfies the system; it is at T_A and the vertical intervals are all intervals 1, 6 or Bs.

In order to review the basic features of a canon system and how its graph is determined, refer to Example 2a. Here we see a fragment of a pc canon generated by a canon system, where intervals P and Q are members of V. The fragment has two voices, voice 1 on the bottom consisting of the notes a, c, e, and voice 2 on the top with notes b, d, f. Voice 2 follows voice 1 at the interval t as shown by the labeled arrow from note a to note b. The adjacent linear intervals between the notes of the voices are the intervals x and y. The vertical intervals of the canon are shown by the capital letters P and Q labeling vertical brackets. Example 2b rewrites 2a with all notes and vertical intervals redefined as functions of the first note of the canon in voice 1 called a, the interval t of the canon, and the linear intervals x and y. The set of equations below determine the values of the vertical intervals P and Q:

$P = (a + t) - (a + x) = t - x$
$Q = (a + t + x) - (a + x + y) = t - y$

Proposition 1. $x = t - P$; $y = t - Q$.

The line marked Proposition 1 shows that the linear intervals in the canon are derived from the vertical intervals (here P and Q) by taking each from t, since $(t - y) - (t - x) = x - y$, we have:

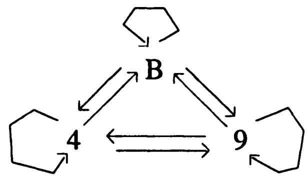
Proposition 2. $Q - P = x - y$ (and $P - Q = y - x$ (by algebra)).

The proposition shows that there is a relationship between each pair of successive vertical intervals (here P and Q) and each associated, successive pair of intervals in a canon voice (here x and y).

We take advantage of Proposition 1 to construct the canon graph for a canon system. Proposition 1 implies that we can derive the set L, the set of linear intervals of the voices for canons that satisfy

Example 1. (a) Canon graph of canon system TCS(A, {1,6,B}); (b) A canon from this system.

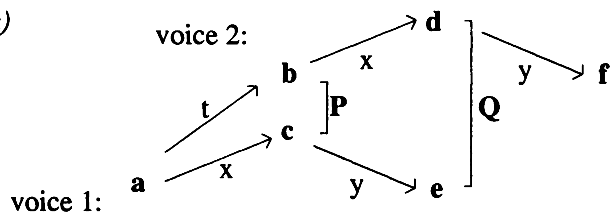
(a)



(b)

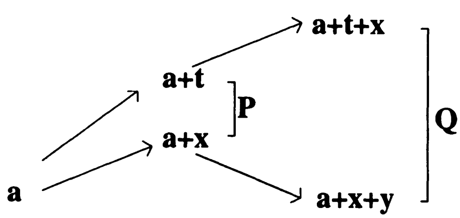
Example 2. (a) Transpositional canon generated by TCS(t, V) where V includes intervals P and Q; (b) Structure of Example 2a.

(a)



a, b, c, d, e, f are pcs; x, y, t, P, Q are directed intervals.

(b)



the system, by taking each member of set V from t. This is written in Proposition 3.

Proposition 3. Given TCS(t, V), for each $v \in V$ there exists an $l \in L$, and $l = t - v$.

In Example 3, we work with the canon system of Example 1a. We derive L from V so that L contains intervals 4, 9, and B. Next we construct a table whose row and column heads each hold a different member of L. The content of a table cell shows a minimum canon whose voices are based on the linear interval sequence x, y, where x is the interval at the cell's row head and y is the interval at the cell's column head. In each cell, the aligned notes between voice 1 and 2 determine the vertical intervals P and Q, members of V. Finally, the set-class content of the minimal canon is given at the bottom of the cell.

Since any interval in V may follow any other, we may move from any cell in the table to any other (appropriately transposing the notes in the canon voices, so that last note of voice 1 in a canon in a cell is the first note of voice 1 in the next cell and so forth). We can rewrite the table as a canon graph, by writing a complete graph for all values of L; this is the canon graph in Example 1a. Since there is no restriction on the order of intervals in L (which implies there is no restriction on the order of the vertical intervals in V), we call this an *unrestricted canon system.*

At this point, it should be clear that a canon system can model any transpositional canon whatsoever—tonal, modal, post tonal, perhaps with a change from chromatic to diatonic intervals and pitch-class. But this degree of generality was not the context of my previous article, for, a tonal or modal canon system is usually restricted so that it will not allow all possible intervals to occur in set L or V. If only consonant vertical intervals are allowed in V, then some of the intervals in L will be omitted, due to the relation between V and L given in Proposition 3. And conversely, if only certain linear intervals are allowed in L, then some vertical intervals in V will be deleted. Moreover, in tonal and modal music, interval sequences in a voice or among successive verticalities are often

Example 3. Canon system table for TCS(A, {1,6,B}).

linear intervals L = {A-1, A-6, A-B} = {4, 9, B}

x and y are successive melodic intervals in voices of canon
P and Q are successive vertical intervals between voices of canon

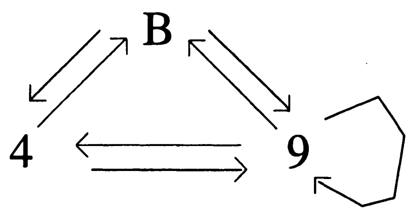
	y = 4	y = 9	y = B
x = 4	V2: A 2 6 V1: 0 4 8 x,y = 4 4 P,Q = 6 6 x-y = Q-P = 0 SC 6-35[02468A]	V2: A 2 B V1: 0 4 1 x,y = 4 9 P,Q = 6 1 x-y = Q-P = 7 SC 6-2[012346]	V2: A 2 1 V1: 0 4 3 x,y = 4 B P,Q = 6 B x-y = Q-P = 5 SC 6-2I[012346]
x = 9	V2: A 7 B V1: 0 9 1 x,y = 9 4 P,Q = 1 6 x-y = Q-P = 5 SC 6-2I[012346]	V2: A 7 4 V1: 0 9 6 x,y = 9 9 P,Q = 1 1 x-y = Q-P = 0 SC 6-23[023568]	V2: A 7 6 V1: 0 9 8 x,y = 9 B P,Q = 1 B x-y = Q-P = A SC 6-2[012346]
x = B	V2: A 9 1 V1: 0 B 3 x,y = B 4 P,Q = B 6 x-y = Q-P = 7 SC 6-2[012346]	V2: A 9 6 V1: 0 B 8 x,y = B 9 P,Q = B 1 x-y = Q-P = 2 SC 6-2I[012346]	V2: A 9 8 V1: 0 B A x,y = B B P,Q = B B x-y = Q-P = 0 SC 5-1[01234]

context sensitive. All of this reduces the allowable sequence of intervals x, y in the canon graph, so it is a subset of the complete graph of an unrestricted canon system. This would seem to imply that tonal and modal systems are subtler and more refined than unrestricted canon systems.

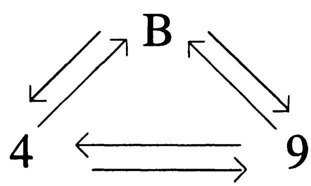
It may be useful to postulate rules that will omit certain sequences in a particular canon system. For example, returning to the canon system table in Example 3, we might not want to use the minimum canon in the bottom right of the table since it has a repetition of the note A, whereas all the other cells have minimal canons that have no pitch-class duplication. If we delete this cell, the linear interval sequence <B,B> will not be permitted and the arrow from B to itself will be deleted from the canon graph. Or one might have an aversion to whole-tone scales, so the cell on the upper left would be deleted, omitting the sequence <4,4> and the arrow from 4 to itself on the graph. Let us omit both of these cells; the resulting canon graph is shown in Example 4a. The new resulting canon system is therefore restricted and context-sensitive. We might omit all the cells on the main diagonal on the table since they involve the immediate repetition of an interval in a canonic voice. Then the canon system would be that of Example 4b. While this restriction could be applied to all canon systems by deleting the main diagonal of cells on a canon system table, there is a particular context-sensitive reason in this case; all the cells not on the main diagonal hold minimal canons that are of the same set-class, 6-2[012346]. Another, more subtle restriction would be to omit cells whose minimal canon content is related to the prime form of 6-2 by inversion, namely the upper right cell, the first cell of row 2, and the middle cell of the bottom row. This restricted graph is shown in Example 4c. We note that one now can only move counter-clockwise among all intervals in the graph. All the restrictions on canon system TCS(A, {1,6,B}) relate to the set-class content of the minimal canons in the cells of the table. This is salient since a canon system can be characterized by the set-classes in the minimum canons on the table. In fact, a canon system can only have as many distinct set-classes as the number of unordered pairs of intervals in V. Thus:

Example 4. Graphs of three restricted canon systems derived from TCS(A, {1,6,B}).

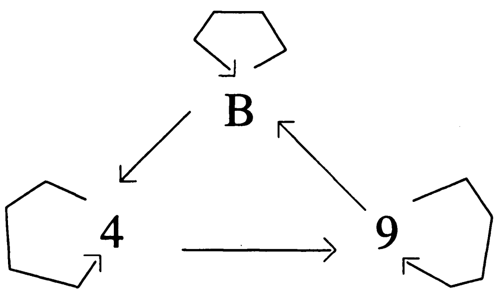
(a)



(b)



(c)



Proposition 4. The maximum number of set-classes represented by the minimal canons of a canon system TCS(t, V) is equal to the number of pairs in V.

This follows because the canon system table is inversionally symmetric around its main diagonal.

Some canon systems have special features. For instance, if for every interval v in a canon system's V-set, the set also contains the inverse of v, the system is said to be *exchangeable.*[3] That means that we can exchange the vertical ordering of the two voices in a canon so the top voice leads the canon, yet still obtain the same vertical intervals. The canon system we have been working with is exchangeable, because it has inverses 1 and B in its V-set as well as 6, its own inverse. We illustrate this invariance of vertical intervals under the exchange of voices in Example 5a. The voices of the canon from Example 1b are exchanged so the top voice leads the canon without altering the set of vertical intervals.

The canons of a particular canon system may be pruned according to the technique illustrated in Example 5b. The example shows two canons from the canon system TCS(A, {3,4,8,9}). The underlined notes of the first canon form a diagonal interval included in the V-set of the canon system. Thus we can delete the note immediately before an underlined note in voice 1 and after an underlined note in voice 2 to produce a new correct canon. The result is the second canon, shown in Example 5c.

Since there are many canon systems with the same V-set, it might be interesting to modulate from one to another, or even embed one canon from one system into that of another. All distinct (unrestricted) canon systems that have the same V-set form what we call a *V-complex*, as defined below:

[3] The traditional term for exchangeable would be "invertible," but using it would confuse inversion as an exchange of voices with inversion as a pitch-class operation.

Example 5. (a) Exchanging voices in a canon derived from an exchangeable canon system; (b) Pruning notes from canons: underlined notes form a diagonal interval included in the V-set of the canon system; (c) A pruned canon derived from Example 5b.

(a)

```
V2:         A 9 6 A 9 1 5
V1:       0 B 8 0 B 3 7
V ints:     B 1 6 B 6 6

V1:       0 B 8 0 B 3 7
V2:         A 9 6 A 9 1 5
V ints:     1 B 6 1 6 6
```

(b)

```
V2:         4 B 5 0 6 7 1
V1:       6 1 7 2 8 9 3
```

(c)

```
V2:         4 5 0 6 1
V1:       6 7 2 8 3
```

Example 6. (a) Canon in TCS(3, V); (b) Canon in TCS(4,V).

(a)

```
V2:  3 A 8 4
V1:0 7 5 1
    7 A 8 = L intervals
     8 5 7 = V intervals
```

(b)

```
V2:  4 0 B 8
V1:0 8 7 4
    8 B 9 = L intervals
     8 5 7 = V intervals
```

Definition 2. A V-complex VC(V) is the set of canons systems TCS(t, V) for all t.

Since there are 12 values for t, each V-complex contains 12 canon systems. The L set of linear intervals associated with each system are related by the addition of a constant. Thus:

Proposition 5. Given two members of a V-complex TCS(r, V) and TCS(r+n, V) and L_r is the L set of TCS(r, V) and L_{r+n} is the L set of TCS(r+n, V), then for each member l of L_r there is a member of L_{r+n} equal to l+n.

Let us construct canon systems TCS(3, V) and TCS(4, V) where V is {5,7,8}. Let r = 3 and r + n = 4, so n = 1. By Proposition 3, the L set of TCS(3, V) is {3-5, 3-7, 3-8} = {A,8,7}; by Proposition 4, the L set of TCS(4, V) is {A+1, 8+1, 7+1} = {B,9,8}. Note that the V intervals in both canons are the same. The members of the L set of the second system are related by the addition of 1 to the members of the first,[4] as shown in Example 6.

As I mentioned above, a composer may be interested in modulating from one canon system to another in the same V-complex, or embedding one in another. I provide one method in Examples 7a-7d. Given canon fragment (1) on the left of Example 7a, the linear interval from a to b is same as the interval from c to d. And the interval from a to c is same as from b to d. We may then exchange pcs in fragment (1) as shown in the example to produce the canon fragment (2) whose vertical interval has the same content as the vertical interval in the original canon fragment. Two verticals are inversionally related. Example 7b illustrates the exchange, but the two fragments may not be from systems within the same V-complex, since their V-sets are not defined to be identical. However, if both v and 12-v are vertical intervals in one fragment, the two fragments may be from canon systems in the

[4] A methodological note: we could say that the L set of TCS(4, V) is the T_1 of the L set of TCS(3, V), and that in general, T_n of the L set of TCS(r, V) is the L set of TCS(r+n, V). However, we should not, since L sets contain intervals, not pitch-classes, and transposition is not defined on intervals.

Example 7. (a) Two canon fragments; (b) Illustration of pc exchange; (c) Modulation between canon systems in V-comples VC({3,4,8,9}); (d) Embedding of one canon in another in V-complex VC({3,4,8,9}).

(a)

```
1)             2)
     c d            b d
   a b            a c
```

(b)

```
V2:    4 6    linear interval = 2; t is 5
V1:  B 1      vertical interval is 3.
              the canon system is TCS(5,{..3..})

V2:    1 6    linear interval is 5; t is 2
V1:  B 4      vertical interval is 9.
              the canon system is TCS(2,{..9..})
```

(c)

```
V2:     6 B 0 1
V1:   9 2 3 4       from canon system TCS(9,{3,4,8,9})

V2:         4 1 B 4
V1:       3 0 A 3   from canon system TCS(1,{3,4,8,9})
```

(d) A canon from canon system TCS(1,{3,4,8,9})

```
V2:     1 B 3 8 1
V1:   0 A 2 7 0
```

embedded by other canons from systems in the V-complex.

```
V2:      1  B       B  3        3  8        8  1
V1:  0  A        A  2        2  7        7  0
     |  |  |     |  |  |     |  |  |     |  |  |
TAQ:    A  B  6  0..2  3  A  5..7  8  9  6..0  1  2  A
Q:   0  1  8  2..A  B  6  1..2  3  4  1..7  8  9  5
     |           |           |           |
 TCS(A,{3,4,8,9})|       TCS(5,{3,4,8,9})|
             TCS(4,{3,4,8,9})        TCS(5,{3,4,8,9})
```

same V-complex. Examples 7c and d show how modulation and embedding work in the V-complex of canon systems with V ={3,4,8,9}.

In analogy to the V-complex, we can also define *L-complexes* of canon systems.

Definition 3. An L-complex is the set of TCS(t, V) that share the same L-set.

The canons in Example 8 share the same L-set. The relation between them is exactly in accord with systems within a V-complex.

Proposition 6. Given two members of an L-complex TCS(r, V_r) and TCS(r+n, V_{r+n}), then for each member v of V_r there is a member of V_{r+n} equal to v+n.

In essence, two canon systems are members of the same L-complex if their first arguments differ by n, and the members of their V-sets also differ by n. There are 12 canon systems in a L-complex. Example 8 provides an instance of two canons from systems in the same L-complex. The last line of the example shows how they are related; n is 1. The example also points out that two canons from different members of an L-complex may be based on the same subject.

Combining all canon systems that are related by virtue of being members of a V- or L-complex produces a *LV-complex* of 144 canon systems:

Definition 4. An LV-complex is the set of all members of the V- and L-complexes of TCS(t, V) that share V and/or L.

Thus two canon systems are LV-related if they share the same V-set or L-set or both. The closure of the LV-complex allows much scope and relatedness among two-voice canons, but it only pertains to two-voice, transpositional, first species canons.

Example 8. (a) Canon from TCS (A, {1,6,B}), with L-set = {4,9,B}; (b) Canon from TCS (B,{2,7,0}, with L-set = {4,9,B}.

(a)
```
V2:         A 7 B A
V1:       0 9 1 0
L ints: 9 4 B
V ints:   1 6 B
```

(b)
```
V2:         B 8 0 B
V1:       0 9 1 0
L ints: 9 4 B
V ints:   2 7 0
```

Note: TCS(B,{2,7,0}) = TCS(A+1,{1+1, 6+1, B+1})

Inversional Canon Systems

As with transpositional canon systems, we can generate inversional canons by the use of canon graphs. Each canon graph is derived from an *inversional pc canon system* (henceforth, i-canon system), denoted by the expression ICS(t, V). Canons that satisfy this system have two voices related by T_tI, whose vertical intervals are limited to members of V.

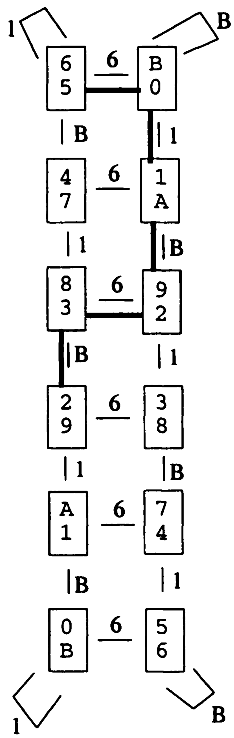
Definition 5. An inversional canon system is denoted by the function ICS(t, V), where t is the inversional index in T_tI, whose vertical intervals are limited to members of V.

Example 9a shows the canon graph for the i-canon system (B, {1,6,B}). The canon in Example 9c is derived from the graph; its two voices are related by T_BI and the vertical intervals are from the system's V-set. The nodes of the graph contain all 12 ordered pairs of notes related by T_BI, and therefore the notes sum to B. The notes are connected by lines labeled by numbers representing intervals drawn from the V-set {1,6,B}. When two nodes immediately connected by a line are chosen to construct a canon, the vertical interval will be the interval labeling the line. Thus, as is the case in transpositional canon graphs, by following the lines we trace paths to generate a particular canon. One such path given by thick lines is shown in Example 9a. Example 9b shows the sequence of nodes visited by the path. The sequence of bottom notes of each node defines voice one of the canon; the sequence of top notes gives voice 2. The canon of Example 9c is produced by shifting the sequence of top notes one position to the right.

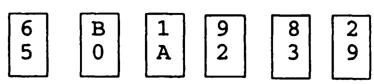
Example 10a reveals the anatomy of an i-canon. The first voice has notes a and c, and the second voice, b and d. Pairs {a,b} and {c,d} are related by T_tI. The interval x spans from a to c; it is reflected by -x spanning from b to d. P is the vertical interval from b to c and is a member of the canon system's V-set. Example 10b shows how the canon fragment in Example 10a is aligned and transformed into a portion of a canon graph; two nodes are connected by a line labeled P.

Example 9. (a) Canon graph of Inversional Canon System (B, {1,6,B}); (b) Path on the canon graph; (c) Resulting canon.

(a)



(b)



(c)

An i-canon graph is made out of all possible canon fragments permitted in a particular i-canon system. In order to construct the graph efficiently, we use an i-canon table. Such a table for constructing graphs for canons generated by ICS(B, V) is found in Example 11a. All pairs of possible canon fragments involving two inversional pairs related by T_BI are arranged in the table. Since there are 12 pairs, there are 144 sequences of two pairs. Each cell holds a unique canon fragment. The canon fragment in a particular cell holds a sequence of two inversional pairs, the first at the head of the cell's row, the second pair at the head of the cell's column. The number in the upper left hand corner of each cell gives the vertical interval in the cell's canon fragment. Note that the cells that hold the same vertical interval are arranged in a diagonal that wraps around the table. In order to use the table, we erase the content of all cells whose canons have vertical intervals not included in the V-set of the canon system. Example 11b shows the table for ICS(B,{1,6,B}); only cells having a vertical interval of 1, 6 or B remain. These are all the canon fragments that can generate canons for this inversional canon system. We construct the graph by starting with one of the canon fragments, which we will call F; we then find other canon fragments that start with the same notes that end F. Since there are three different intervals in the V-set, there will be three different fragments on the table that start with the last notes of F. This follows from the fact that each row and column of the table has three filled-in cells. This discussion implies the next two propositions:

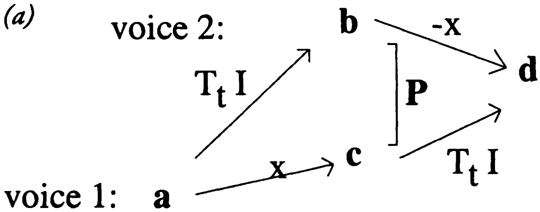
Proposition 7. In a canon graph of ICS(t, V), there are 12 nodes, each of which is connected by n lines to other nodes, where n is the cardinality of V.

Proposition 8. In an i-canon, there is no mapping from the intervals in the V-set to the linear intervals in the voices.

Proposition 7 suggests that i-canon graphs of canon systems with V-sets larger than four members are difficult to display in two dimensions without many crossing lines and complex configurations. Proposition 8 is illustrated by any i-canon table, where many i-canon fragments involving different linear intervals

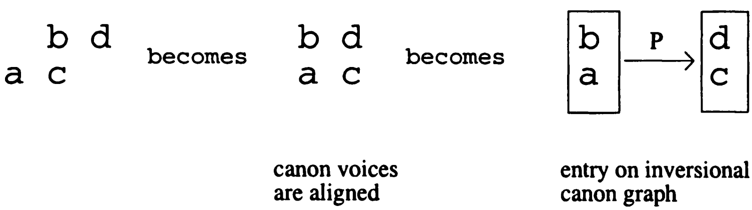
Example 10. (a) Inversional canon generated by ICS(t, V) where V includes interval P; (b) Generating a part of an inversional canon graph from the canon fragment in Example 10a.

(a)



a, b, c, d are pcs; x, -x, P are directed intervals; t is the inversional index.

(b)



Example 11a. Table for inversional canon systems ICS(B, V).

	B 0	A 1	9 2	8 3	7 4	6 5	5 6	4 7	3 8	2 9	1 A	0 B
B 0	B BB 00	A BA 01	9 B9 02	8 B8 03	7 B7 04	6 B6 05	5 B5 06	4 B4 07	3 B3 08	2 B2 09	1 B1 0A	0 B0 0B
A 1	A AB 10	9 AA 11	8 A9 12	7 A8 13	6 A7 14	5 A6 15	4 A5 16	3 A4 17	2 A3 18	1 A2 19	0 A1 1A	B A0 1B
9 2	9 9B 20	8 9A 21	7 99 22	6 98 23	5 97 24	4 96 25	3 95 26	2 94 27	1 93 28	0 92 29	B 91 2A	A 90 2B
8 3	8 8B 30	7 8A 31	6 89 32	5 88 33	4 87 34	3 86 35	2 85 36	1 84 37	0 83 38	B 82 39	A 81 3A	9 80 3B
7 4	7 7B 40	6 7A 41	5 79 42	4 78 43	3 77 44	2 76 45	1 75 46	0 74 47	B 73 48	A 72 49	9 71 4A	8 70 4B
6 5	6 6B 50	5 6A 51	4 69 52	3 68 53	2 67 54	1 66 55	0 65 56	B 64 57	A 63 58	9 62 59	8 61 5A	7 60 5B
5 6	5 5B 60	4 5A 61	3 59 62	2 58 63	1 57 64	0 56 65	B 55 66	A 54 67	9 53 68	8 52 69	7 51 6A	6 50 6B
4 7	4 4B 70	3 4A 71	2 49 72	1 48 73	0 47 74	B 46 75	A 45 76	9 44 77	8 43 78	7 42 79	6 41 7A	5 40 7B
3 8	3 3B 80	2 3A 81	1 39 82	0 38 83	B 37 84	A 36 85	9 35 86	8 34 87	7 33 88	6 32 89	5 31 8A	4 30 8B
2 9	2 2B 90	1 2A 91	0 29 92	B 28 93	A 27 94	9 26 95	8 25 96	7 24 97	6 23 98	5 22 99	4 21 9A	3 20 9B
1 A	1 1B A0	0 1A A1	B 19 A2	A 18 A3	9 17 A4	8 16 A5	7 15 A6	6 14 A7	5 13 A8	4 12 A9	3 11 AA	2 10 AB
0 B	0 0B B0	B 0A B1	A 09 B2	9 08 B3	8 07 B4	7 06 B5	6 05 B6	5 04 B7	4 03 B8	3 02 B9	2 01 BA	1 00 BB

Example 11b. Table for ICS(B, {1,6,B}).

	B 0	A 1	9 2	8 3	7 4	6 5	5 6	4 7	3 8	2 9	1 A	0 B
B 0	B BB 00					6 B6 05					1 B1 0A	
A 1					6 A7 14					1 A2 19		B A0 1B
9 2				6 98 23				94 27	1 93 28		B 91 2A	
8 3			6 89 32					1 84 37		B 82 39		
7 4		6 7A 41					1 75 46		B 73 48			
6 5	6 6B 50					1 66 55		B 64 57				
5 6					1 57 64		B 55 66					6 50 6B
4 7				1 48 73		B 46 75					6 41 7A	
3 8			1 39 82		B 37 84					6 32 89		
2 9		1 2A 91		B 28 93					6 23 98			
1 A	1 1B A0		B 19 A2					6 14 A7				
0 B		B 0A B1					6 05 B6					1 00 BB

have the same vertical interval. Because we cannot define an L-set for a given i-canon system, we cannot define an L-complex of i-canon systems. But it is possible to define V-complexes for i-canon systems in analogy to those of transpositional systems. However, t may only be changed by adding an even number to it. Thus, ICS(t, V) and ICS(t+n, V) are members of the same V-complex if n is even. The reason for this is well known in atonal theory. The operations T_tI and $T_{t+n}I$ are conjugates of each other and therefore have isomorphic structural properties. The twelve T_tI operators therefore fall into two groups of six operations whose ts are related by the addition of an even number. Thus we have Propositions 9 and 10:

Proposition 9. There are two V-complexes of i-canon systems: V(o, V) and V(e, V) where o is odd and e is even.

Proposition 10. The canon graphs of i-canon systems ICS(t, V) and ICS(t+2x, V) are isomorphic. The pcs in the nodes of the latter system are those of the former each transposed by x. (The labels of the graph's lines [the intervals in the V-set] are unchanged.)

Proposition 9 states that there are only two distinct V-complexes of i-canon systems; t is either odd or even. Proposition 10 specifies when i-canon systems are isomorphic. Proposition 10 has two consequences: (1) we need only construct canon graphs for ICS(B, V) and transpose them by x to get the canon graphs for ICS(B+2x, V); (2) there are only two canon tables necessary to generate all i-canon systems and graphs: the one in Example 11a for canons at T_BI, and one for canons at T_0I (or some other even t in T_tI).

Proposition 11. All i-canons are dual. If either voice leads the canon, the vertical intervals are preserved.

This proposition defines duality. All i-canons are dual; whether the top voice or bottom voice leads the canon, the vertical intervals are preserved. Duality is illustrated in Example 12a. Transpositional canons do not have duality. Inversional canons may be exchangeable under the same criterion for transpositional canons.

Example 12. (a) Duality in an i-canon with V-set intervals preserved; (b) Exchange and duality in a canon derived from ICS(B, {2,3,8,9}).

(a) ICS(3, {3,8,B})

```
V2:        3 B 0 6 5     V2:     3 B 0 6 5
V1:      0 4 3 9 A       V1:       0 4 3 9 A
V ints:  B 8 3 8         V ints: B 8 3 8
```

(b) ICS(B, {3,4,8,9}) is exchangeable.
Thus, vertical intervals are preserved in all four canons.

original canon:

```
V2:        B 9 6 8
V1:      0 2 5 3
V-ints:  9 4 3
```

exchange:

```
V1:      0 2 5 3
V2:        B 9 6 8
V-ints:  3 8 9
```

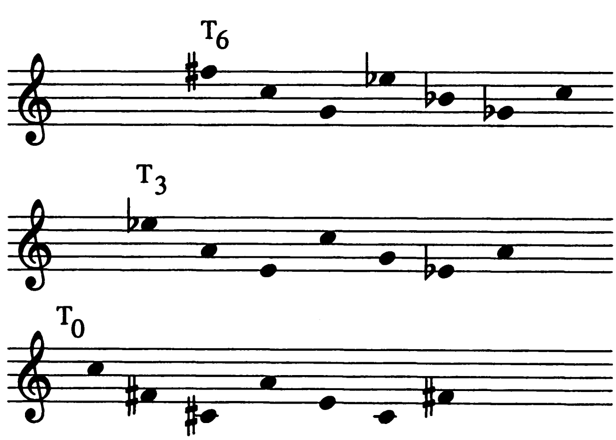
duality:

```
V2:      B 9 6 8
V1:        0 2 5 3
V-ints:  9 4 3
```

duality and exchange:

```
V1:        0 2 5 3
V2:      B 9 6 8
V-ints:  3 8 9
```

Example 13. A three-voice stacked canon at T_3 with every three-note vertical a member of 3-11[037].



If so, all reconfigurations of the two voices preserve the vertical intervals specified by the i-canon system. See Example 12b.

Stacked Canons

Let us now turn to what is the subject of the last part of this paper, the generation of what have been called "stacked canons" by Alan Gosman.[5] Such canons are first-species canons in any number of voices related by transposition by n; that is, voice v is related to the first by T_{nv}.

Definition 6. A stacked canon is a first-species canon in any number of voices related by transposition by n, that is, voice v is related to the first by T_{nv}.

While Gosman put forward a method to generate stacked canons in Renaissance music, we describe a new method to generate post-tonal stacked canons that have members of a select group of set-classes among the verticals.[6] Example 13 shows a canon of this type; the interval between successive voices is T_3, the verticals are members of SC 3-11[037]—major and minor chords. Such canons are made by beginning with a *generating string* as shown in Example 14. We describe the generating string A, as the array A_0, A_1, A_2, ... A_z. To make the stacked canon we construct a canon at the unison based on the generating string A. We then successively transpose each vertical of the generating canon by T_n successively from left to right to produce the *resultant canon*; each mth vertical of the generating canon is transposed by T_{mn}. Voice 1 of the resultant canon now differs from A and is called the *subject string*. Voice 2 of the resultant canon is the T_n of the subject string; voice k is the $T_{n(k-1)}$ of the subject string. Example 15 documents how the resultant canon of Example 13 was constructed from its generating string and canon.

[5] See Gosman 1997.

[6] This method was suggested and partially implemented by Nathan Schmidt.

Example 14. Schema for constructing a stacked canon.

Generating string A = A_0, A_1, A_2, A_3 ... A_z

Generating canon:

V3:			A_0	A_1	...	A_{z-2}	A_{z-1}	A_z
V2:		A_0	A_1	A_2	...	A_{z-1}	A_z	
V1:	A_0	A_1	A_2	A_3	...	A_z		
col. #	C_0	C_1	C_2	C_3	C_5–C_{z-1}	C_z	C_{z+1}	C_{z+2}

Subject string = T_0A_0, T_nA_1, $T_{2n}A_2$, $T_{3n}A_3$ ... $T_{zn}A_z$

Resultant canon:

V3:			T_2A_0	T_3A_1	...	$T_{nz}A_{z-2}$	$T_{n(z+1)}A_{z-1}$	$T_{n(z+2)}A_z$
V2:		T_1A_0	T_2A_1	T_3A_2	...	$T_{nz}A_{z-1}$	$T_{n(z+1)}A_z$	
V1:	A_0	T_1A_1	T_2A_2	T_3A_3	...	$T_{nz}A_z$		
col. #	T_0C_0	T_nC_1	$T_{2n}C_2$	$T_{3n}C_3$	$T_{4n}C_4$–$T_{z(n-1)}C_{z-1}$	$T_{nz}C_z$	$T_{n(z+1)}C_{z+1}$	$T_{n(z+2)}C_{z+2}$

Example 15. Constructing the canon in Example 13.

Generating string = 0 3 7 0 4 9 0

Generating canon:

V3:			0	3	7	0	4	9	0
V2:		0	3	7	0	4	9	0	
V1:	0	3	7	0	4	9	0		
col. #	C_0	C_1	C_2	C_3	C_4	C_5	C_6	C_7	C_8

Subject string = $T_0$0, $T_3$3, $T_6$7, $T_9$0, $T_0$4, $T_3$9 $T_6$0 = 0 6 1 9 4 0 6

Resultant canon:

V3:			6	0	7	3	A	6	0
V2:		3	9	4	0	7	3	9	
V1:	0	6	1	9	4	0	6		
col. #	C_0	T_3C_1	T_6C_2	T_9C_3	T_0C_4	T_3C_5	T_6C_6	T_9C_7	T_0C_8

V2 = T_3(V1); V3 = T_3(V2) = T_6(V1)

Returning to Example 14 we note that each vertical in the unison canon based on the generating string A is a segment of A. The set-classes of imbricated elements of array A form the verticals in the resultant canon because its verticals are transpositions of the verticals of the generating canon. Since the transposition interval n is free, the generating string will form a V-complex of stacked intervals, all with the same set-classes in each corresponding column. From Example 14 we see that the set-classes of the verticals in the resultant canon depend on the structure of the generating string. If that generating string imbricates only one set-class, K, each column of the resultant canon will have K as its set-class. The generating string of Example 13 (as shown in Example 15) has such a structure, so each three-note successive segment is a member of K, which is set-class 3-11[037].

So, in order to construct stacked canons that have only a few different set-classes in their verticals, we have to construct generating strings that have only a few different imbricated set-classes. There are many ways to do this. Such strings can be generated by the LPR neo-Riemannian transformations—or equivalently—by flipping triangles on the *Tonnetz*, or—also equivalently—by following a path on a two-partition graph. In a recent paper,[7] I showed how the *Tonnetz* can be transformed so that the "triangles" on the resulting *Tonnetz* are of set-classes other than 3-11[037]. On the top left of Example 16 we see the classical *Tonnetz* with ic 4 as verticals and ic 3 as horizontals and triangles with NW-SE diagonals containing set-class 3-11[037]. This *Tonnetz* is transformed via rotation of its second and third row into the *Tonnetz* on the right. The resulting *Tonnetz* has ic 5s as verticals and ic 3s as horizontals, and triangles with NW-SE diagonals containing set-class 3-7[025]. Triangle flips on any *Tonnetz* preserve two notes and the set-class membership of the trichord. Repeated triangle flips produce a chain of notes imbricated by the trichords of the *Tonnetz*. We can use this chain as a generating string to produce a stacked canon, which will have verticals only belonging to the set-class of the triangles on the *Tonnetz*. The bottom portion of Example 16 shows a series of triangle flips and

[7] Morris 1998.

Example 16. Transforming the Tonnetz *and finding a generating string on it.*

the *Tonnetz*
vertical = ic 4
horizontal = ic 3

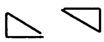
C	Eb	F#	A
E	G	Bb	Db
G#	B	D	F

rotate 1x →

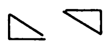
→ rotate 2x

transformed *Tonnetz*
vertical = ic 1
horizontal = ic 3

C	Eb	F#	A
G	Bb	C#	E
D	F	G#	B

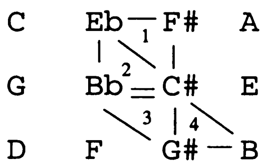
triangles 

hold members of 3-11[037]

triangles 

hold members of 3-7[025]

Triangle flips on the transformed *Tonnetz.*

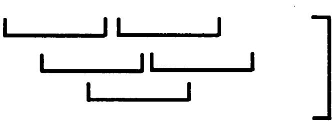


successive triangle flips:

triangle 1 (E♭ F♯ B♭) is flipped to
triangle 2 (E♭ B♭ C♯) is flipped to
triangle 3 (B♭ C♯ G♯) is flipped to
triangle 4 (C♯ G♯ B).

The content of the successive triangle flips yields the generating string A which has imbricated trichords from only the set-class 3-7[025]:

A = F♯ E♭ C♯ B♭ G♯ C♯ B



all members of 3-7[025]

the resulting generating string A, which imbricates members of set-class 3-7[025]. The method of triangle flips is similar to the generation of what David Lewin called *Riemannian Systems.*[8] Each system is a string based on orderings of trichords that do not have inversional symmetry. Such strings have the structure displayed in Example 17.[9]

Any of the elements A_i of the generating string A can be segments of pcs; we henceforth call them cells. Thus we can have strings that will generate stacked canons of second species and others, such as that in Example 18a. Note that columns now replace the verticals of the first-species canons. We can use Lewin's definition of a Riemannian System on the segments in each cell of A. The result will be a series of imbricated set-classes (larger than trichords) of the same class. This is only one way to generate strings imbricated by one or a few set-classes.[10] When such strings are rows, they are called super saturated set-type rows.[11]

Examples 18a-18c show what results if the generating strings of stacked canons are super saturated set-type rows. In Example 18a, the generating string is a row of this type. Each group of four successive cells in the generating string is a member of 6-14[013458]. The generating canon therefore has the five imbricated members of this set-classes as its middle columns. The subject string is a rotated, transposed, MI-form of the generating string, so it is also a row with the same properties.

The generating string of Example 18b is a super saturated row involving the Z-related set-classes 6-28 and 6-49, so the middle columns of the resultant array are members of these set-classes. But the subject string is not a row; it has duplications and its set-class is 8-9[01236789]. Since the periodicity of the T_2 operator is equal to the number of cells in the subject string, the canon can be wrapped

[8] See Lewin 1982.

[9] Lewin's definition of a Riemannian System may be generalized by letting each A_i be a string of pitches and A_i and $A_{(i+1)}$ are successive strings in A.

[10] This topic was first discussed in Morris 1982-3.

[11] Super saturated set-type rows are studied in Morris 1983-4.

Example 17. Properties of Lewin's Riemannian Systems.

Each Riemannian system is a series of pitch-classes A = A_0, A_1, A_2, A_3, A_4, A_5, etc.

A has the properties:

($A_y \cup A_{(y+1)}$) where y is even is a transposition of ($A_0 \cup A_1$)

($A_y \cup A_{(y+1)}$) where y is odd is a transposed inversion of ($A_0 \cup A_1$)

Therefore, ($A_y \cup A_{(y+1)} \cup A_{(y+2)}$) for all y, is a member of the same set-class.

Example 18. (a) Generating string imbricating members of 6-14[013458]; (b) The generating string imbricates members of Z-related set-classes 6-28 and 6-49; (c)Rotational array based on Example 18b.

(a)

0 3	4	8 5	1	9 6	A	B 2	7

Resultant canon: T_3 stacked canon based on subject string; verticals are all members of 6-14[013458]:

V4:				9 0	4	B 8	7	6 3	A	3 5	1
V3:			6 9	1	8 5	4	3 0	7	B 2	A	
V2:		3 6	A	5 2	1	0 9	4	8 B	7		
V1:	03	7	2 B	A	9 6	1	5 8	4			

In this case, the subject string is the r_6T_9MI of the generating string.

(b)

0 1	4 7	8 A	B 2	5 6	9 3

Resultant canon: T_2 stacked canon based on subject string; verticals are members of 6-28 and 6-49:

V3:			4 5	A 1	4 6	9 0	5 6	B 5
V2:		2 3	8 B	2 4	7 A	3 4	9 3	
V1:	0 1	6 9	0 2	5 8	1 2	7 1		

(c)

V3:	4 5	A 1	4 6	9 0	5 6	B 5
V2:	8 B	2 4	7 A	3 4	9 3	2 3
V1:	0 2	5 8	1 2	7 1	0 1	6 9

around to form the rotational array shown in Example 18c; its columns are all members of 6-28 and 6-49.[12] We see that rotational arrays are implicated in stacked canons.

Proposition 12. Let A be a stacked canon generated by operation K, whose subject string has c cells, and v_c voices. If c or v_c = the periodicity of K, A can be converted into a rotational array.

The proposition asserts that a stacked canon can be converted into a rotational array if the canon's generating operation has a periodicity equal to the number of voices in the canon or equal to the number of cells in the subject string.[13] The mention of the canon's generating operation refers to the transposition operation applied successively to the voices, but as we will see, we can generate stacked canons under other operations.

Example 19 makes two points about the generating string. First, if one has a stacked canon, one can determine its generating string by operating on its subject string.

Proposition 13. Let G be a generating string of a stacked canon under successive transpositions by t; H is a subject string. Let G_n and H_n be the nth cell of G and H, respectively. $H_n = T_{tn}G_n$ and $G_n = T_{-tn}H_n$

Example 19 shows that different interpretations of the generating string can produce multiple subject strings. Since the generating string can be composed of cells, not only individual notes, the cells can hold sets or be empty. This can be useful in making canons where the resulting subject string might be "uninteresting" if one took the generating string as a series of pcs. For instance, in Example 19, the generating string is 0A31 and the canon interval is T_2, and the resulting subject string is 0077. If we wanted a subject string with no repetitions for a canon under T_2, we could

---

[12] See Rogers 1967 and Morris 1988.

[13] The periodicity of an operation is the number of times it has to be performed to equal the identity operator. For transposition operators the periodicity is 12/t, if t divides 12, or 12.

Example 19. Alternate interpretations of generating strings.

V6:		53				A8
V5:	31				86	
V4:				64		1B
V3:			42		B9	
V2:		20		97		
V1:	0A		75			

Compressed into three voices at T_4:

V3:	31	53			86	A8
V2:			42	64	B9	1B
V1:	0A	20	75	97		

Example 20. Inverse stacked canons.

023 = G; t = 1; c = 5 g = 3, v_c = 3

Canon A:

```
035
 146
  257
```

Canon B

320 = R(G) t = B; c = 5; n = 1(4)

```
31A
 209
  1B8
```

B = RT_4A written upside-down and backward.

reinterpret the generating string as (0A| |31); then the subject string would be (0A| |75). The new substring generates rotational arrays with each column a member of the set-class of the generating string, set-class 4-10[0235]. In this way, the composer has a choice of subjects if the canon operator is fixed in advance.

Proposition 14. Let A be a stacked canon of c columns under successive transpositions by t with the generating string G. Let B be a stacked canon of c columns under successive transpositions by -t with the generating string RG. A = T_nB written upside down and backward where n is t(c-1). A and B are inverse canons. Note, c = g + v_c-1, where g is the number of cells in G and v_c is the number of voices in the canon.

This proposition asserts that every stacked canon has an inverse; if we have a stacked canon A under the operation T_t based on generating string G and write it backwards and upside-down and transpose it by a certain transposition operator, then the result, B, is a stacked canon under T_{-t} based on the retrograde of G. Thus, canon B is the inverse of A, and vice versa. Example 20 provides an example of a canon and its inverse.

Here we generalize stacked canons to any operator and its cycle group. Repeated iterations of an operator, such as T_n, produce a cyclic group.[14]

Proposition 15. Generalized stacked canons are constructed by operating on the cells of a generating string by the members of a cyclic group generated by the operation X. Let A_n be the nth cell of the generating string, and B_n the corresponding cell of the subject string. Then $B_n = X^nA_n$, and voice k of the resulting canon is $X^{n(k-1)}$ of the subject string.

The expansion to all operators, of course, will produce stacked canons where voice m will be related to voice m-1 by the operator

[14] The members of the group are X, the generator, XX or X^2, XXX or X^3, until $X^n = T_0$, the identity operator. Thus, n is the periodicity of the operator and the order of the group.

Example 21. (a) A stacked canon with $X = T_5I$ and $X^2 = T_0$; (b) Two stacked canons based on the cyclic group generated by the operator T_1M; the members of the group are (T_1M), $(T_1M)2 = T_6$, $(T_1M) = T_7M$, $(T_1M)4 = T_0$.

(a)

```
Generating string: 0 1 B 3 8 A 4 9 7 6 2 5
Subject string:    0 4 B 2 8 7 4 8 7 B 2 0
```

V3:			0	4	B	2	8	7	4	8	7	B	2	0
V2:		5	1	6	3	9	A	1	9	A	6	3	5	
V1:	0	4	B	2	8	7	4	8	7	B	2	0		
SCs			3-1 [012]	3-6 [024]	3-11 [037]	3-9 [027]	3-8 [026]	3-5 [016]	3-7 [025]	3-2 [013]	3-4 [015]	3-3 [014]		

(b) Two voice canon:

Generating string = 01127189I9A (content of voice ∈ 8-6[0123789A]); imbricated cells ∈ 4-6[0127]

Subject string = 011B0I23I89

Two voice canon; each column ∈ 4-6[0127]

V2: T_1MVc1		16	81	B4	A3
V1:	01	B0	23	9A	

A four voice canon on the same subject string as above but with $v_c = 4$ = periodicity of group

The canon is a rotational array; each column ∈ 8-6[0123789A].

Voices 1 and 3, and 2 and 4 are related by T_6.

V4: T_7MVc1	27	5A	49	70
V3: T_6Vc1	89	34	67	56
V2: T_1MVc1	A3	16	81	B4
V1:	01	B0	23	9A

X; however, if the operation is neither transposition nor inversion then the set-class content of adjacent cells in the generating string of a stacked canon will not be preserved in a column of the stacked canon. If we use some T_tI operation as the generator, the stacked canon will only have two transforms of the subject within it, as the periodicity of any T_tI operation is 2. Example 21a generates a three-voice canon under T_5I, so that the first and third voices of the canon are identical. The canon has ten different trichords in its columns since the generating string is a ten-trichord row.[15] A cyclic group of order four generated by the T_1M operation generates stacked canons in Example 21b; both canons have the same subject string, and the first is embedded in the second. The generating string imbricates set-class 4-6[0127] and its entire content is a member of the complementary set-class, 8-6[0123789A]. These two set-classes contain sets that are invariant under T_nM, so the content of the columns of the canons does not change set-class affiliation under the T_1M or T_7M operations. Of course, the canons appear to have two different subject strings, but they are related under the operations in the group, which include T_6.

By way of conclusion, it should be clear that pc canons are related to other branches of atonal theory, namely generalized voice-leading, rotational arrays, transpositional combination, compositional spaces and design,[16] and Neo-Riemannian theory. This article has identified striking differences between canons based on transposition and other operations, and in particular, inversion. This is just another demonstration that transposition and inversion, the two generators of the classical set-class group, are quite different and need not be given equal footing in theory and analysis. Perhaps the main contribution of this paper shows that the vertical and horizontal dimensions of pitch-class function can be interrelated in a strict canonic environment, especially in the case of stacked canons, where the horizontal relations in the generating string are found in the vertical columns of the canon.

---

[15] Babbitt 1987 refers to this kind of row as an all-trichord row.

[16] Canon graphs are compositional spaces and canons are compositions designs. See Morris 1998.

References

Babbitt, Milton. 1987. *Words About Music.* Ed. Stephen Dembski and Joseph N. Strauss. Madison: University of Wisconsin Press.

Gosman, Alan. 1997. "Stacked Canon and Renaissance Compositional Procedure." *Journal of Music Theory* 41.2: 289-317.

Lewin, David. 1982. "A Formal Theory of Generalized Tonal Functions." *Journal of Music Theory* 26.1: 23-61.

Morris, Robert. 1982-3. "Combinatoriality Without the Aggregate." *Perspectives of New Music* 21.1-2: 187-217.

_____. 1983-4. "Set-Type Saturation Among Twelve-Tone Rows." *Perspectives of New Music* 22.1-2: 432-86.

_____. 1988. "Generalizing Rotational Arrays." *Journal of Music Theory* 32: 204-45.

_____. 1997. "The Structure of First-Species Canon in Modal, Tonal and Atonal Musics." *Intégral* 9: 33-66.

_____. 1998. "Voice Leading Spaces." *Music Theory Spectrum* 20.2: 175-208.

Rogers, John. 1967. "Toward a System of Rotational Arrays." *Proceedings of the American Society of University Composers* 2: 61-84.