Spatial Computing for Musical Transformations and Counterpoint

Louis Bigo & Jean-Louis Giavitto & Antoine Spicher

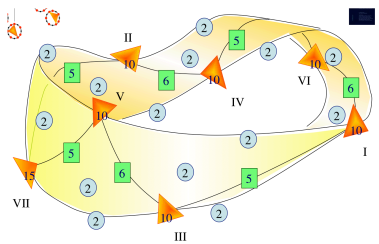
mgs.spatial-computing.org

LACL, Université Paris Est Créteil – IRCAM

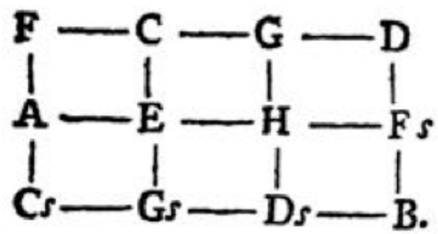
Spatial Computing Workshop 2013

Saint Paul Minnesota USA – May 6th - 10th 2013

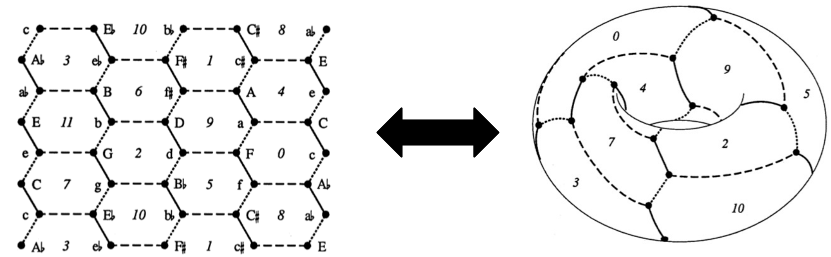
Spaces for musical representations



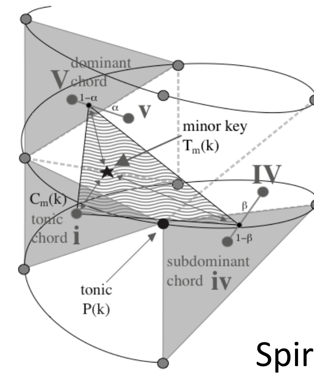
Tonality strip [Mazzola]



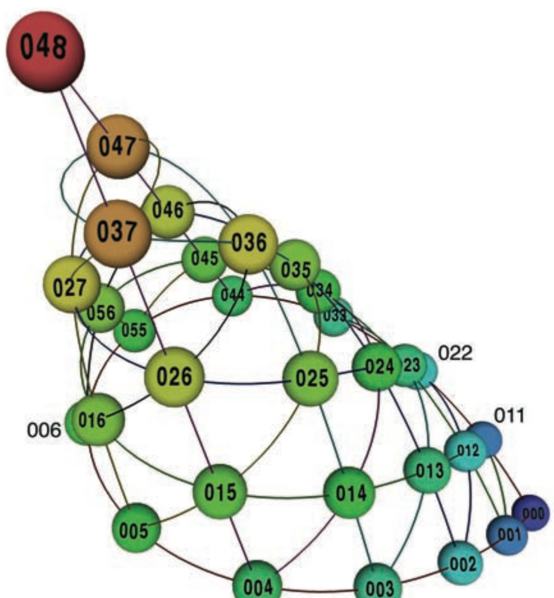
Speculum Musicum [Euler]



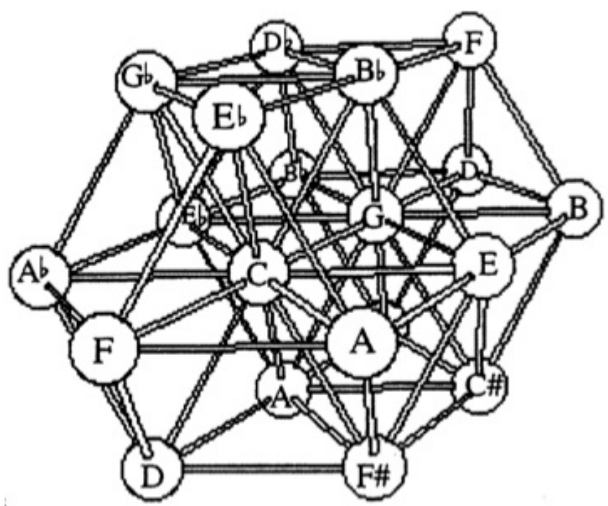
Chicken Wire Torus [Douthett & Steinbach]



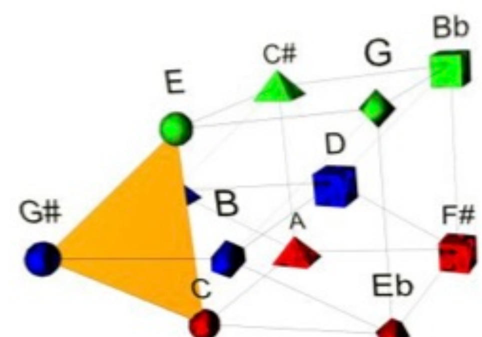
Spiral Array [Chew]



Orbifolds [Tymoczko]



3D Tonnetz [Gollin]



Model Planet [Barouin]

Context

- Musical theory and computer science
- *Spatial computing* for musical theory and analysis
  - Compute in space
  - Compute space
- MGS
  - *Unconventional programming* language for spatial computing

    Intuitive (natural) way to express computations on/in space
  - Introduction of topological concepts in a programming language
  - Two main principles
    - Space: *topological collection*
    - Computation: *transformation*

Outline

- Space and Collection of Chords
- Applications
  - Harmonization
  - Geometrical Transformations in Chord Spaces
  - Spatial Counterpoint
- Conclusion

Outline

- Space and Collection of Chords
- Applications
  - Harmonization
  - Geometrical Transformations in Chord Spaces
  - Spatial Counterpoint
- Conclusion

Spatial Representation of Chords

- Chords in Music

  A collection of notes played "simultaneously"

  - A set of pitches (event on a staff)

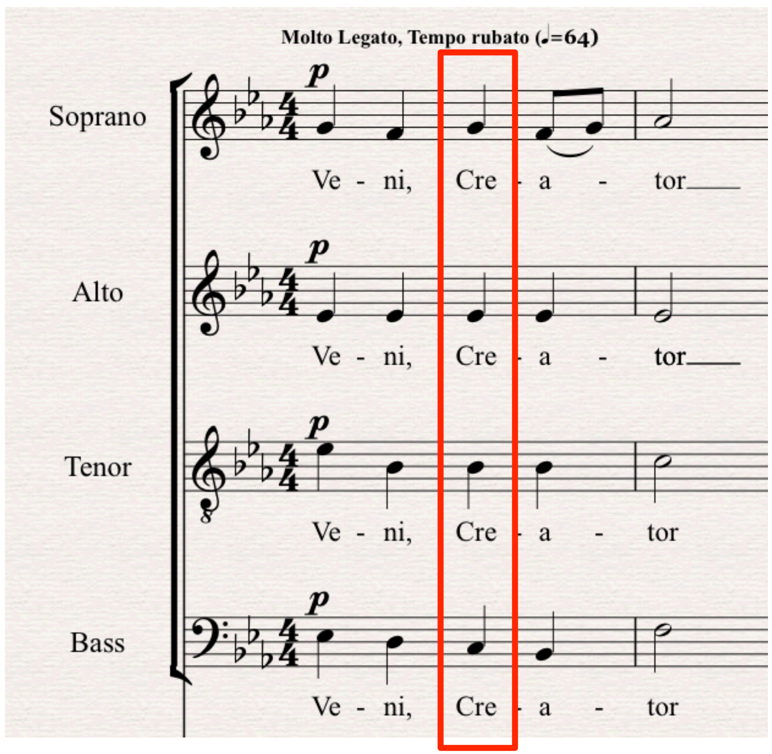
{ E3, E4, D4, G#4, B4 }

Spatial Representation of Chords

- Chords in Music

  A collection of notes played "simultaneously"

  - A set of pitches (event on a staff)
  - A sequence of pitches (choral voices)



[C3, Bb4, Eb4, G4]

Spatial Representation of Chords

- Chords in Music

  A collection of notes played “simultaneously”

  - A set of pitches (event on a staff)
  - A sequence of pitches (choral voices)
  - An ordered set of pitch classes (chord progression)

```
  A7     A7     A7     A7
|////|////|////|////| the first 4 bars

  D7     D7     A7     A7
|////|////|////|////| the second 4 bars

  E7     D7     A7     E7
|////|////|////|////| the third 4 bars
```

A7: { tonic = A, third = C#, fifth = E, seventh = G# }
D7: { tonic = D, third = F#, fifth = A, seventh = C# }
E7: { tonic = E, third = G#, fifth = B, seventh = D# }

Spatial Representation of Chords

- Chords in Music

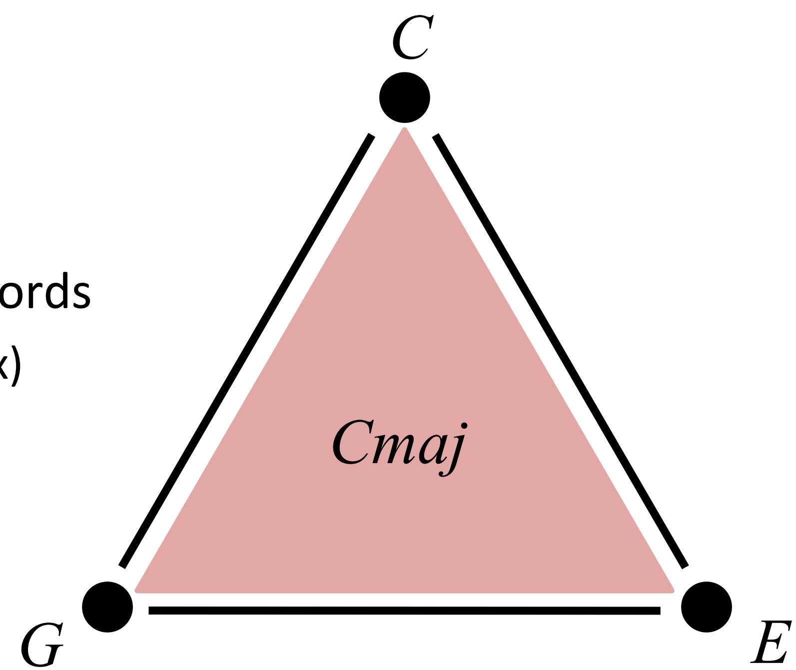
  A collection of notes played "simultaneously"

  - A set of pitches (event on a staff)
  - A sequence of pitches (choral voices)
  - An ordered set of pitch classes (chord progression)
  - A **set of pitch classes** (*Set Theory* in music)
  - …

- *p*-Chords as (*p* - 1)-simplexes

  Representation of a chord and all its subchords

  - 1-chord (a pitch class): 0-simplex (vertex)
  - 2-chord: 1-simplex (edge)
  - 3-chord: 2-simplex (triangle)



Spatial Representation of Chords

- Chords in Music

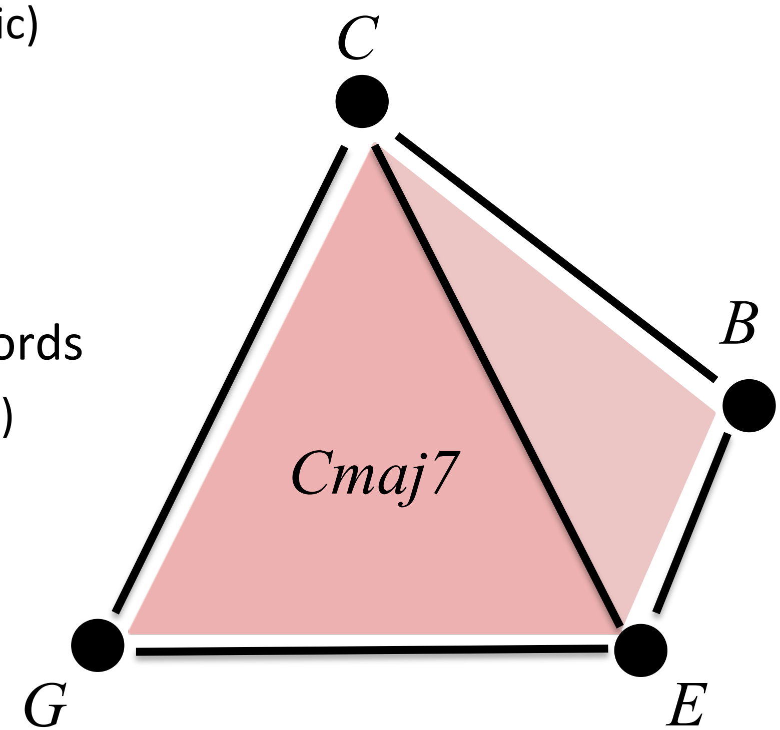
  A collection of notes played "simultaneously"

  - A set of pitches (event on a staff)
  - A sequence of pitches (choral voices)
  - An ordered set of pitch classes (chord progression)
  - A **set of pitch classes** (*Set Theory* in music)
  - …

- *p*-Chords as (*p* - 1)-simplexes

  Representation of a chord and all its subchords

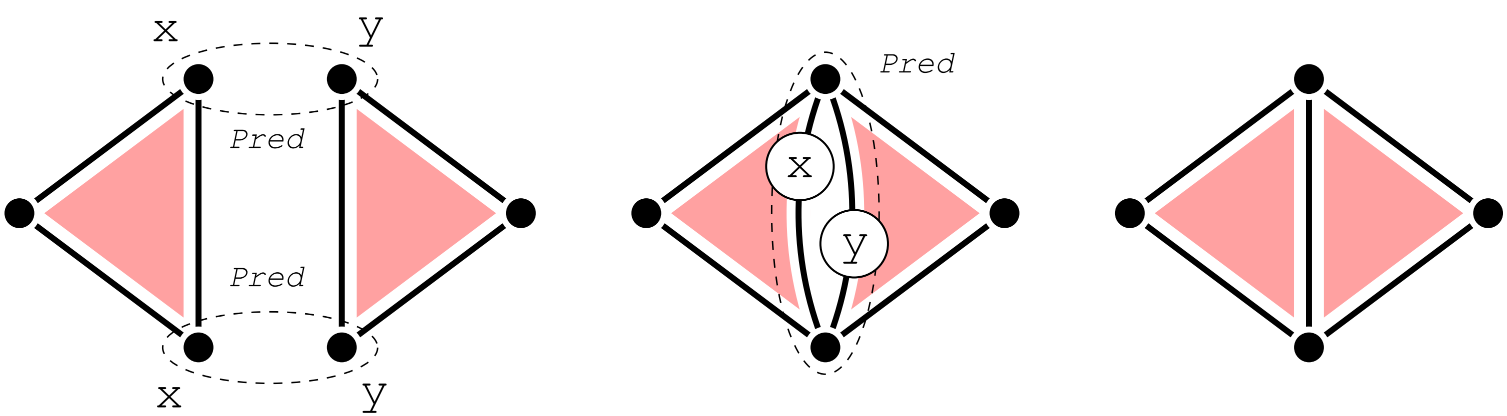
  - 1-chord (a pitch class): 0-simplex (vertex)
  - 2-chord: 1-simplex (edge)
  - 3-chord: 2-simplex (triangle)
  - 4-chord: 3-simplex (tetrahedron)
  - …



Building a Chord Space

- Self-assembly of cellular complexes

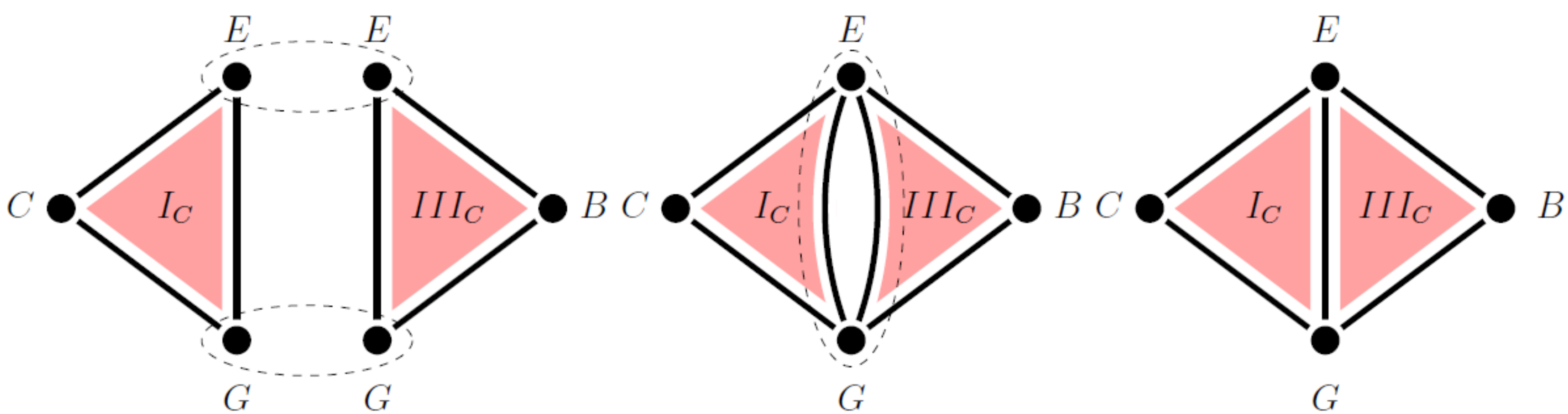
  Reaction of the sub-complexes between themselves




```
Trans Self-Assembly[Pred, Label] = {
  x y / (Pred x y) and (faces x = faces y)
    =>
      let c = new_cell (dim x)
                       (faces x)
                       (union (cofaces x)
                              (cofaces y))
      in (Label x y) * c
}
```

Building a Chord Space

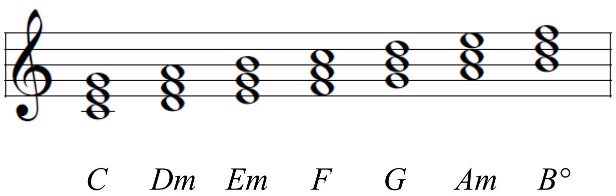
- Application of transformation *Self-Assembly*
  - Basic elements: a population of chords
  - Assembly predicate (***Pred***): same pitch-class subset
  - New label (***Label***): the pitch-class subset



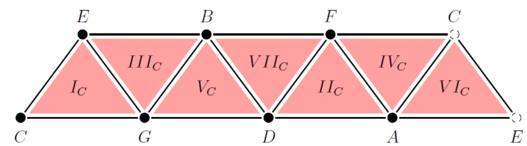

```
Pred x y = (x=y)
Label x y = x
```

Examples: Degrees of a Tonality

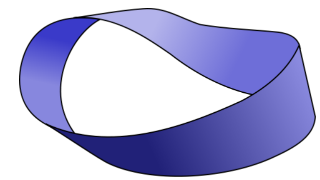
- Degrees of the diatonic scale



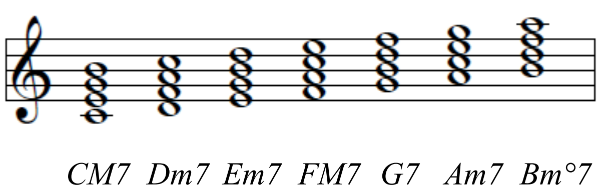
→



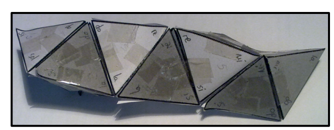
→



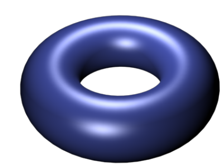
- Tonality with four note degrees



→



→

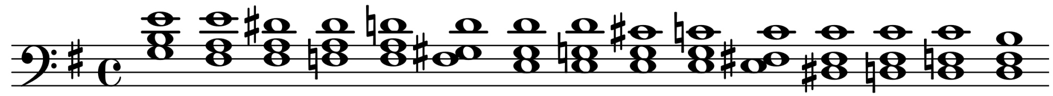


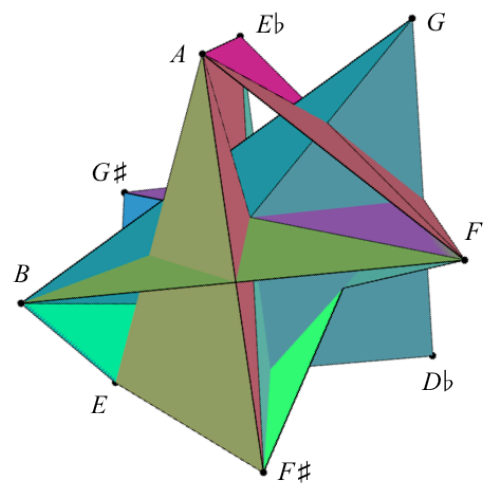
[SCW10]

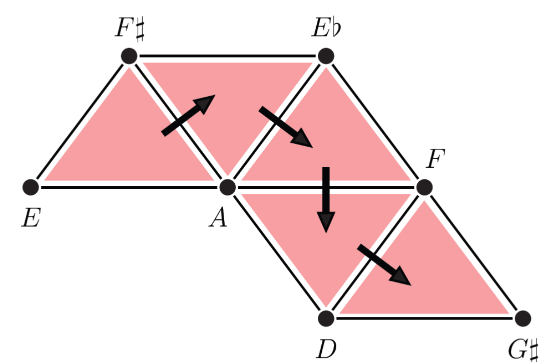
Examples: Chords of a Musical Piece

- Extract of the Prelude No. 4 Op. 28 of F. Chopin

→







120 (2,1)-Hamiltonian paths

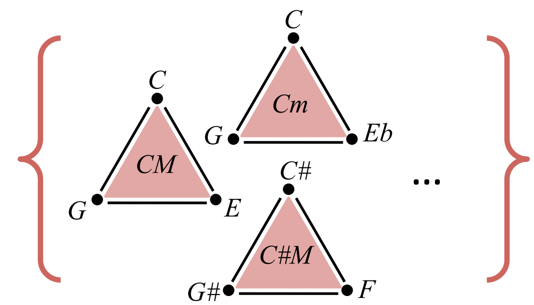
[SCW10]

Examples: Chord Classes

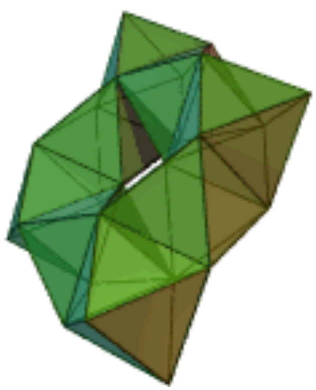
+ Inversion
Transposition

Chord Class
(*i.e.*, a chord and a set of operations)

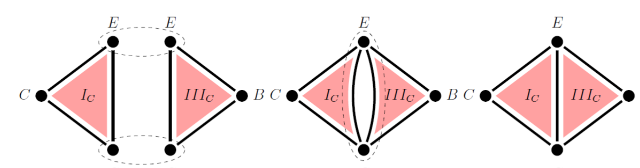
↔



Simplicial Representation
(24 minor and major triads)



Class Complex



Self-assembly Process

Examples: Chord Classes

- Chord classes modulo inv./trans. for 3-chords

$C(n_1, n_2, n_3)$	Simplicial Complex	Representative Trichord
$C(1,1,10)$	cylinder	$\{0,1,2\} = \{C, C^\sharp, D\}$
$C(1,2,9)$	torus	$\{0,1,3\} = \{C, C^\sharp, D^\sharp\}$
$C(1,3,8)$	torus	$\{0,1,4\} = \{C, C^\sharp, E\}$
$C(1,4,7)$	torus	$\{0,1,5\} = \{C, C^\sharp, F\}$
$C(1,5,6)$	circle of 6 tetrahedra boundaries	$\{0,1,6\} = \{C, C^\sharp, F^\sharp\}$
$C(2,2,8)$	two disjoint cylinders	$\{0,2,4\} = \{C, D, E\}$
$C(2,3,7)$	torus	$\{0,2,5\} = \{C, D, F\}$
$C(2,4,6)$	two disjoint circles of 3 tetrahedra boundaries	$\{0,2,6\} = \{C, D, F^\sharp\}$
$C(2,5,5)$	cylinder	$\{0,2,7\} = \{C, D, G\}$
$C(3,3,6)$	three disjoint tetrahedra boundaries	$\{0,3,6\} = \{C, E^\flat, G^\flat\}$
$C(3,4,5)$	torus	$\{0,3,7\} = \{C, E^\flat, G\}$
$C(4,4,4)$	four disjoint 2-simplices	$\{0,4,8\} = \{C, E, G^\sharp\}$

- Chord classes modulo inv./trans. for *n*-chords

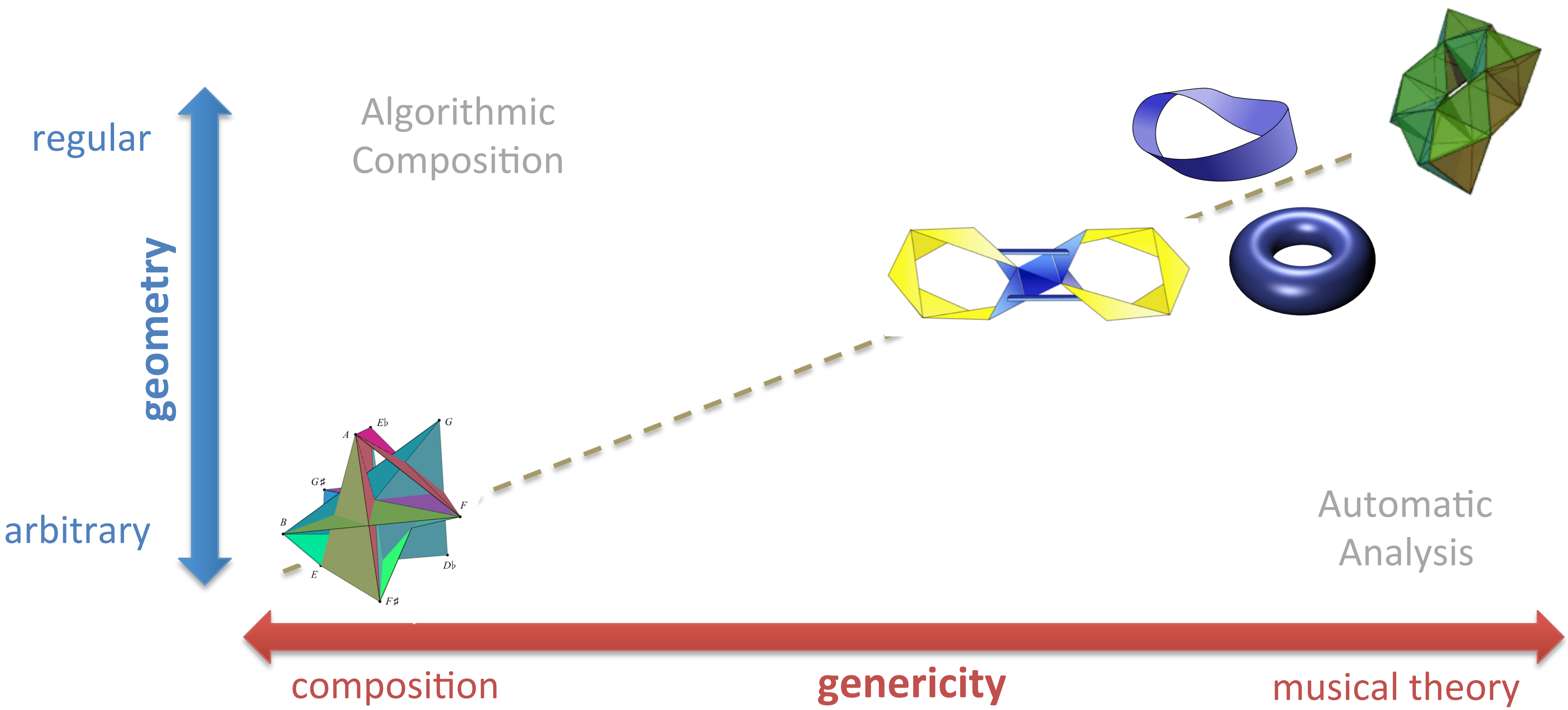
  224 class complexes

Classification of Chord Spaces

- Which chord space for which application?

  Two dimensions

  - Regularity of the geometry
  - Origin of the initial population of chords



Outline

- Space and Collection of Chords
- Applications
  - Harmonization
  - Geometrical Transformations in Chord Spaces
  - Spatial Counterpoint
- Conclusion

Outline

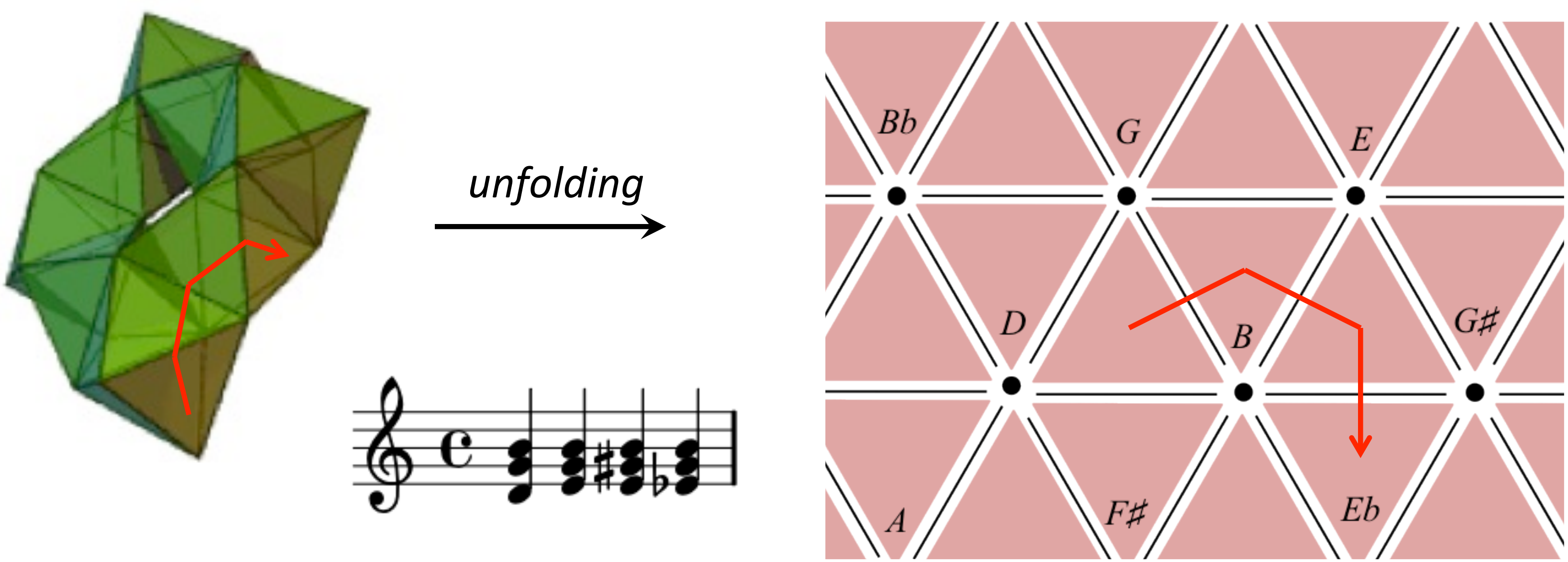
- Space and Collection of Chords
- Applications
  - Harmonization
  - Geometrical Transformations in Chord Spaces
  - Spatial Counterpoint
- Conclusion

Geometrical Transformation of Musical Space

- Motivation
  - Spatial representations of chord classes
    - Strong algebraic support
    - Regular geometry
  - Geometrical transformations
    - Function from a space to itself preserving the structure
    - Translation, rotation, scaling, etc.
  - *Musical meaning of such geometrical transformations?*
- Proposition
  - Application to chord classes modulo inversion/transposition
  - Three steps work
    - Unfolding of a class complex
    - Trajectory computation
    - Transformation application

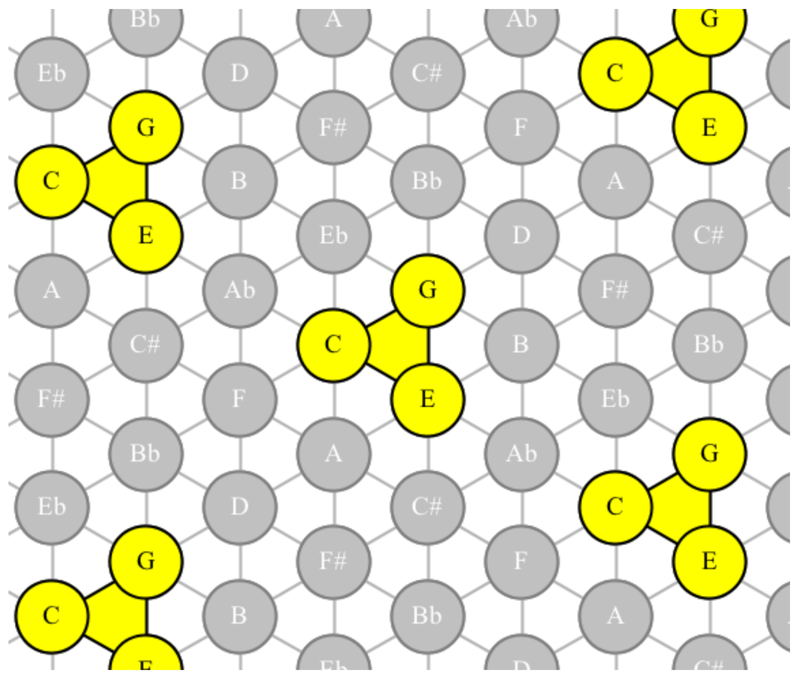
Step 1: Unfolding

- Unfolding of Class Complexes
  - Folding of an infinite hexagonal grid
  - Natural embedding in E^2
  - Local conservation of neighborhood relationships



Step 2: Trajectory of a computation

- Main Drawback of the unfolding
  - Multiple (infinite) locations for the same object
    - Infinite possible representations of a chord sequence
  - Some equivalent locations can be transformed in non-equivalent locations
    - E.g., rotations keeps the center unchanged but moves other instances



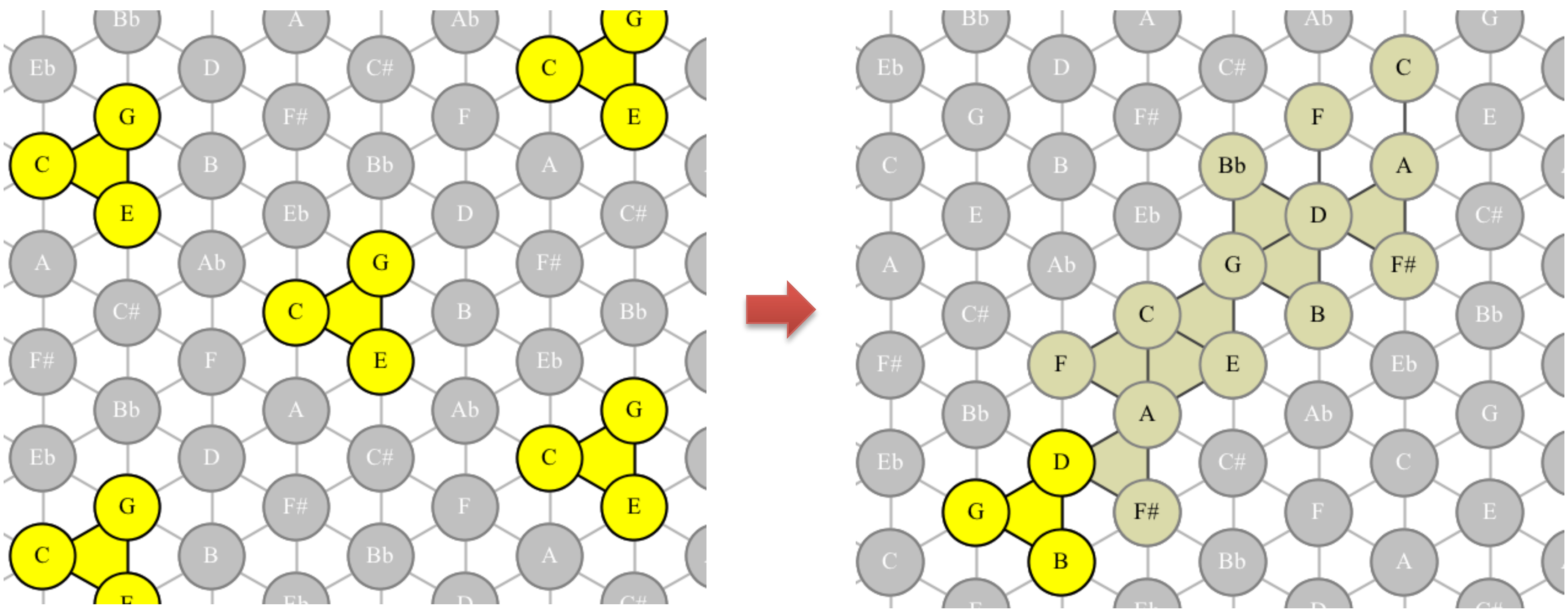
Step 2: Trajectory of a computation

- Main Drawback of the unfolding
  - Multiple (infinite) locations for the same object
    Infinite possible representations of a chord sequence
  - Some equivalent locations can be transformed in non-equivalent locations
    E.g., rotations keeps the center unchanged but moves other instances

- Trajectory generation algorithm
  - Principles
    - Start from an arbitrary position
    - For each chord, choose the closest position from the previous one
  - Visualization of a chord sequence
    - Based on the notion of neighborhood
    - Respect to the underlying chord class

Step 2: Trajectory of a computation

- Main Drawback of the unfolding
  - Multiple (infinite) locations for the same object
    - Infinite possible representations of a chord sequence
  - Some equivalent locations can be transformed in non-equivalent locations
    - E.g., rotations keeps the center unchanged but moves other instances

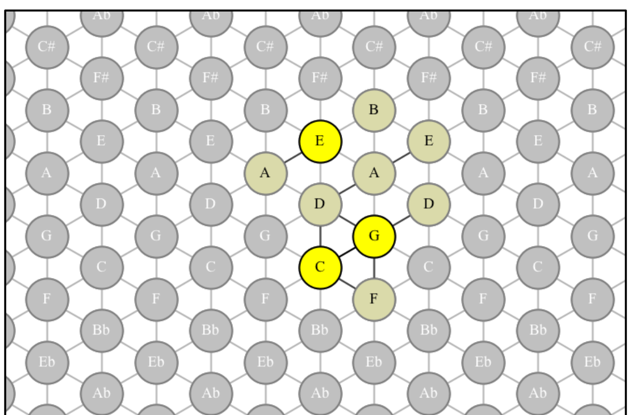


Step 2: Trajectory of a computation

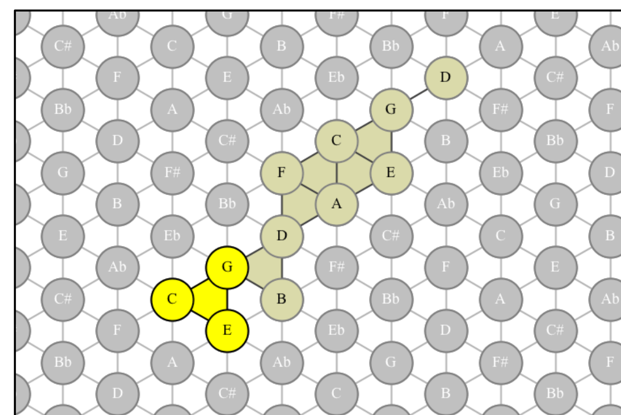
- Main Drawback of the unfolding
  - Multiple (infinite) locations for the same object
    - Infinite possible representations of a chord sequence
  - Some equivalent locations can be transformed in non-equivalent locations
    - E.g., rotations keeps the center unchanged but moves other instances

J.-S.Bach - Choral BWV 256

C(2,5,5)



C(3,4,5)

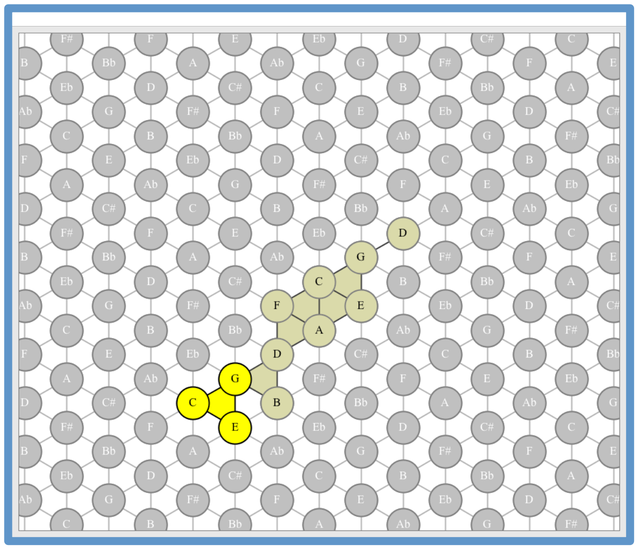


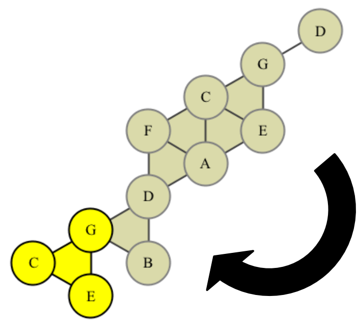
Step 3: Geometrical Transformation

- In the same class complex

  Translation (= transposition) & rotation

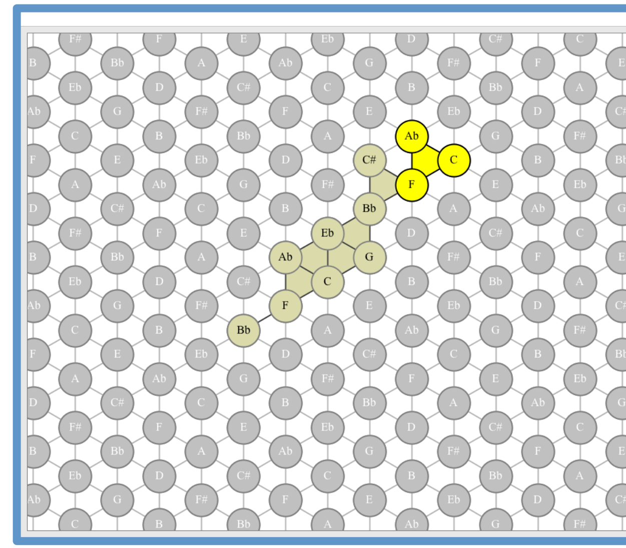
C(3,4,5)

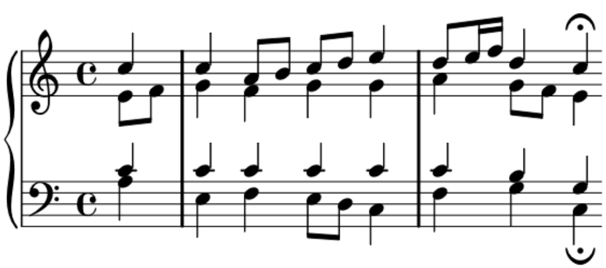




$\xrightarrow{R(\pi)}$

C(3,4,5)

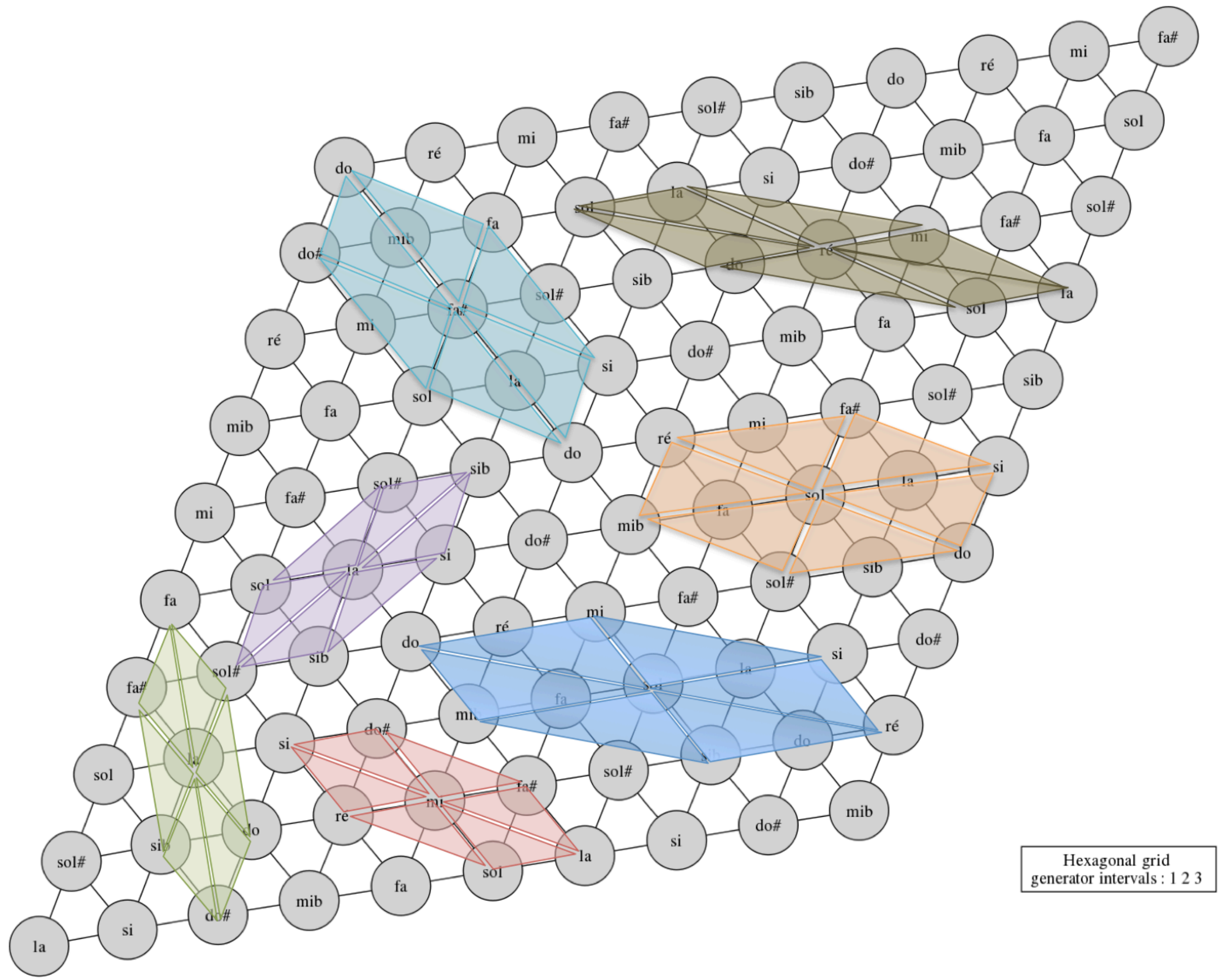




Step 3: Geometrical Transformation

- From a class complex to another

  Scaling

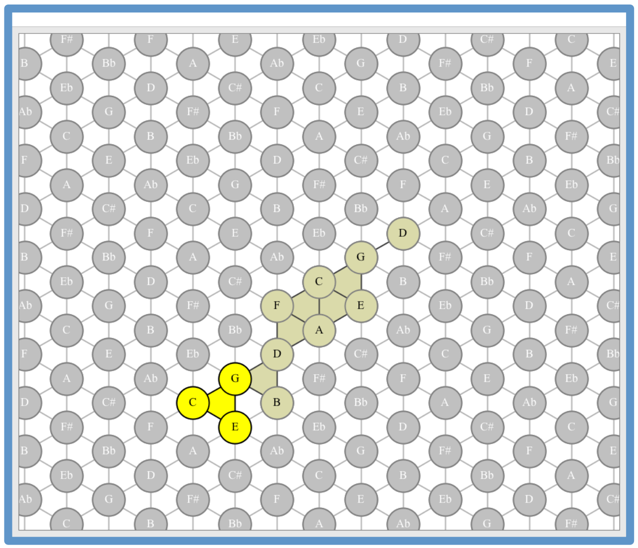


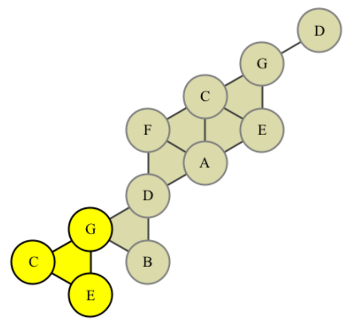
Step 3: Geometrical Transformation

- From a class complex to another

  Scaling

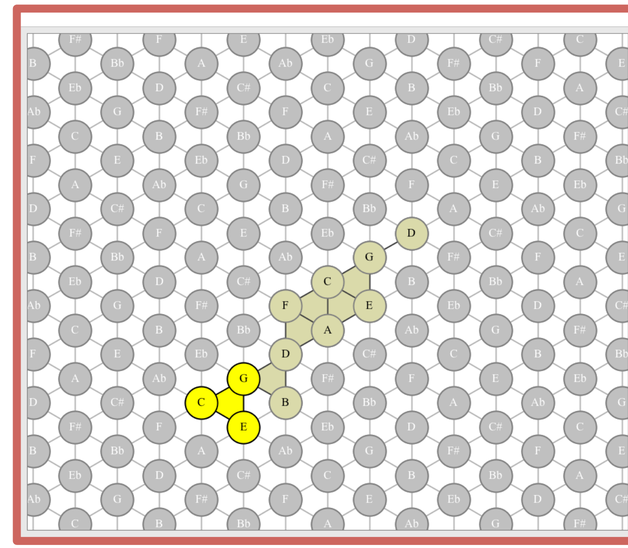
C(3,4,5)





→

C(2,3,7)



Step 3: Geometrical Transformation

- Some audio results

original	
$R(\pi)$ in $C(3,4,5)$	
$T(1,-2)$ in $C(1,2,4)$	
$R(\pi)$ in $C(1,2,4)$	

W.A. Mozart
Piano Sonata No. 16 - Allegro

original	
$R(\pi)$ in $C(3,4,5)$	
$R(2\pi/3)$ in $C(3,4,5)$	
$C(2,3,7)$ ↓ $C(1,2,4)$	

C. Corea
Eternal Child

original	
$R(\pi)$ in $C(3,4,5)$	
$T(1,-2)$ in $C(1,2,4)$	
$C(1,2,9)$ ↓ $C(1,2,4)$	

The Beatles
Hey Jude

Step 3: Geometrical Transformation

- Musical interpretation of some spatial transformations

Trajectory transformation	Musical meaning
Translation in a chromatic space	Transposition
π – Rotation in a chromatic space	Inversion
Translation in a diatonic space	Modal transposition
π – Rotation in a diatonic space	Modal Inversion
ϕ – Rotation in a chromatic space (with $\phi \neq \pi$)	?
ϕ – Rotation in a diatonic space (with $\phi \neq \pi$)	?
Transformation of the underlying space	?

Outline

- Space and Collection of Chords
- Applications
  - Harmonization
  - Geometrical Transformations in Chord Spaces
  - Spatial Counterpoint
- Conclusion

Conclusion & Perspectives

- Spatial Computing for Musical Purpose
  - Use spatial representations of musical objects
  - Neighborhood, locality ⇔ Musical Property
  - Contribution to spatial computing

    Abstract symbolic spaces, not only a population of devices (personal opinion)

- Current and Future Developments
  - Others applications
    - Harmonization (generation of extra-voices in a choral from spatial constraints)
    - Counterpoint rules rephrased in spatial terms
  - Feedbacks with musicologists and composers
  - Tools (Hexachord, PAPERTONNETZ)
  - Short term rendez-vous: Louis' PhD defense!

Questions?

Acknowledgements
Olivier Michel (UPEC), Moreno Andreatta, Carlos Agon, Jean-Marc Chouvel, Mikhail Malt (IRCAM)