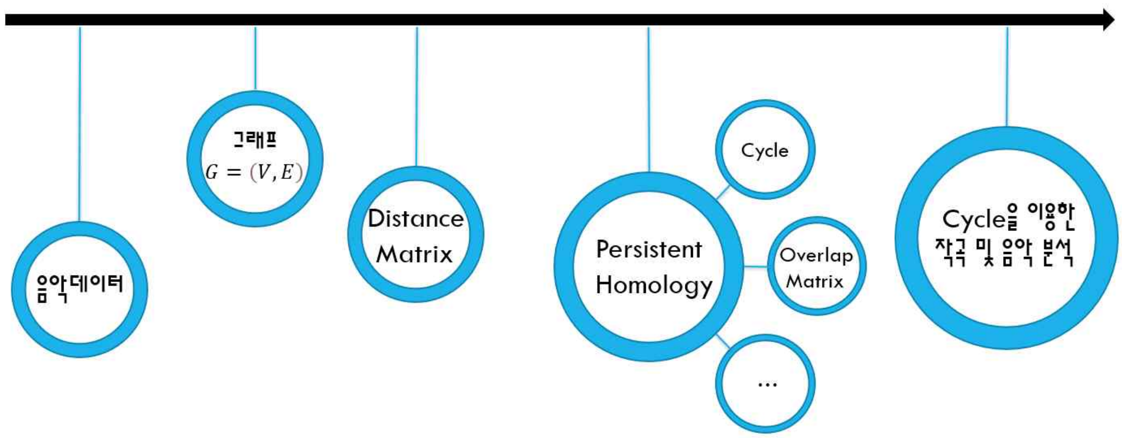
# 국소적 대역적 Cycle 분석 방법론과 합주 분석에의 적용




POSTECH 허은우

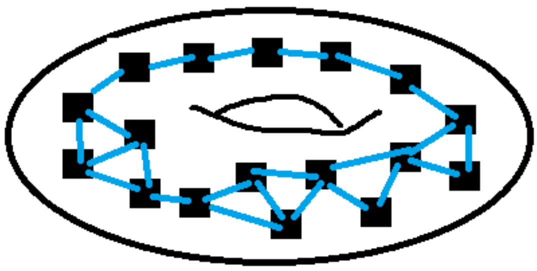
## 1. 위상수학적 데이터 분석기법을 이용한 음악 분석의 전반적 과정
## - 대역적, 국소적 분석 방법론의 소개

인공지능(AI)의 발달이 이루어지고 있는 시대의 큰 흐름에 발맞추어 예술, 교육 분야 등 다양한 분야에 걸쳐 적극적인 연구가 이루어지고 있다. 이에 더불어, 위상수학적 데이터 분석기법을 이용하여 한국음악을 분석하는 시도가 이루어지고 있다. 그렇다면, 과연 현재 어떻게 위상 수학을 이용하여 한국음악을 분석하는 시도를 하고 있을까? 위상수학적 데이터 분석을 이용해 음악을 분석하는 전체적인 큰 흐름에 대해 알아보자.

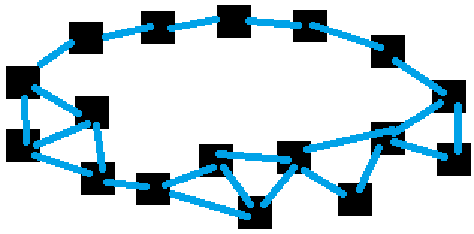
먼저, 음악데이터로부터 점과 변으로 이루어진 그래프를 그린다. 이를 뮤직네트워크(Music Network)라 하고 기호로는 보통 $G=(V,E)$로 표시한다. 여기서 $V$는 점들의 집합이고, $E$는 변들의 집합이다. 그렇게 뮤직네트워크 $G$를 그리고 나면, 이제 임의의 두 점(vertex)에 대한 거리(distance)를 구할 수 있게 된다. 이를 행렬로 표현한 것을 바로 거리행렬(the distance matrix)라 한다. 이제, 임의의 두 점 사이의 거리를 알게 되면 Persistent Homology 기법을 적용 할 수 있게 된다. 기존의 점 집합에 대해서, 거리행렬의 낮은 값에 해당하는 두 점을 차례로 이어가며 변(edge)을 만들고, 삼각형이 생기면 면(face)을 칠하는 방식을 차례대로 진행한다. 진행하다 보면 그래프에서 Cycle이라고 하는 것들이 발견되고 하는데, 우리는 이러한 Cycle을 이용하여 여러 가지 음악 분석에 이용한다. 예를 들면, 새로운 곡의 작곡에 Cycle을 이용하고, 또는 곡의 숨은 구조를 찾는 데도 이용 할 수 있다.

위상 수학의 기본 개념은 고대부터 있었지만, 본격적인 발전은 19C 후반부터 20C 초반에 펠릭스 클라인, 앙리 푸앵카레 등의 저명한 수학자들로부터 이루어졌다. 위상 수학의 철학은 우리가 분석하고자 하는 대상의 위상적 개념을 찾아내는 것이다. 여기서 '위상적 개념'의 예시에는 여러 가지가 있는데, 대표적인 예시로는 물체의 구멍 개수가 있다. 예를 들어 1개의 손잡이가 있는 컵과 도넛이 있다고 하자. 여기서 우리가 계산하고자 하는 '구멍의 개수'에서는, 어떤 물체가 가지고 있는 구체적인 크기와 색 또는 구체적인 생김새 등은 관심이 없다. 컵은 손잡이 부분에서 구멍 1개가 되고, 도넛은 중심의 구멍 1개가 되어 결국 컵과 도넛은 구멍의 개수가 1개로 같다. 따라서, 위상 수학에서는 1개의 손잡이의 컵과 구멍 1개의 도넛은 '구멍의 개수' 관점에서는 같은 대상으로 판별한다. 이외에도 위상적 개념에는 다양한 예시가 있을 것이다. 그런데, 이러한 위상 수학적 관점이 우리에게 주는 이점은 무엇일까? 인간은 기본적으로 혼합된 정보를 받아들이면 왜곡해서 받아들이는 경향이 있다. 다시 말하면, 정보가 너무 많으면 그 정보들은 우리가 알고 싶은 것에 대해 도움을 준다기보다 오히려 방해되는 경우가 많다. 이는 착시현상과 같은 대표적인 현상으로부터 쉽게 생각해볼 수 있다. 위상 수학에서는 어떤 물체의 위상적 특징을 알아내고자 할 때, 그 특징 외에 다른 것들을 모두 걷어내고 오로지 알고자 하는 특징에 집중하려고 시도한다. 예를 들어 컵과 도넛을 비교할 때, 구멍의 개수를 세고 싶은 관점에서는 컵과 도넛의 크기와 색과 같은 성질들은 오히려 관찰자에게는 혼란을 주는 정보라고 할 수 있다. 이러한 시도는 비단 위상 수학뿐만 아니라 모든 수학 분야에서 추구하는 철학이라고 볼 수 있다. 따라서, 위상 수학을 제대로 알기 위해서는 이러한 질문을 던져볼 수 있을 것이다. "위상 수학에서는 어떤 정보를 무시하고, 어떤 정보를 알고 싶어 하는 것인가?"

Persistent homology는 어떤 대상을 차례로 구성해나가며 (예를 들면, 점과 변이나 면을 하나씩 추가해 나가며) 매 순간 일어나는 위상적 성질들의 변화를 알아보는 학문이다. 순차적 과정에서 구멍이 어느 순간 생겼다가 어느 순간 없어지기도 한다. Persistent homology에서는 이렇게 '구멍(hole)'과 같은 위상적 정보가 생겼다가 사라지는, 즉 지속(persistent)되는 현상을 관찰한다. 만약 아래의 그림과 같이, 구멍이 1개인 도넛에 많은 점을 찍고 변들을 짧은 거리의 점들부터 차례로 변을 연결한다고 해보자. 그러면 '거리'가 특정 값에 이르렀을 때, 그림과 같이 변들이 Cycle을 이루어 하나의 고리를 형성하는 순간이 올 것이다. 거리의 값이 충분히 커진다면 모든 점이 이어져서 만들어졌던 Cycle이 없어질 것이다(삼각형이 만들어지면 변의 구분이 없어진다고 여기는 규칙이 있다).



그렇다면 이제, 우리는 도넛 위에 점을 찍는 것이 아니라, 반대로 도넛 없이 데이터가 점으로만 표현되어 있다고 해보자. 그러면 우리는 도넛 없이 Persistent homology 기법을 이용하면 그 데이터들이 이루는 위상적 물체가 1개의 구멍(hole)을 가진 도넛이라는 것을 어느 정도 추정할 수 있다.

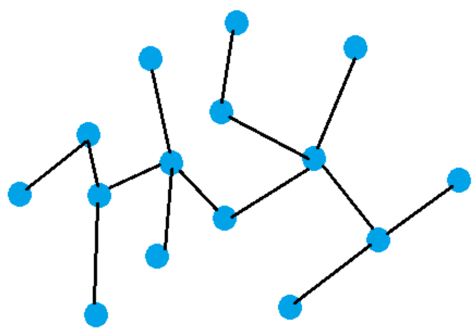


즉, 우리는 데이터로부터 그 데이터가 이루는 위상적 구조를 Persistent homology 기법을 이용하여 '추정'해 볼 수 있다. 데이터의 위상적 구조를 추정해 보는 과정에서 출현하는 Cycle은 어떻게 보면 그 데이터가 이루는 위상구조의 대표적 특징을 말해주는 개념이라 할 수 있다. 따라서 Cycle은 더욱 특별한 의미를 지닌다. 이러한 관점에서, 우리는 음악이 주어졌을 때 각 점을 (Pitch, Duration)으로 표현을 한 뒤, 연속 연주되는 두 점 사이의 변(edge)에 연주가 이루어질 때마다 값(weight of edge)을 1씩 늘려가는 방법으로 그래프 $G=(V,E)$를 그린다. 그리고 그 점 집합 $V$에서 Persistent homology 기법을 적용하였을 때 생기는 Cycle을 이용하여 여러 가지 음악 분석을 시도한다.

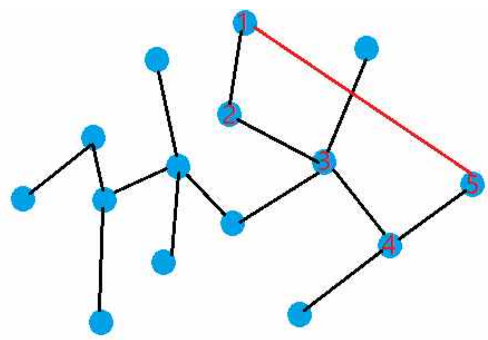
그런데 Persistent homology에서 Cycle을 다룰 때, 조금 짚고 넘어가야 할 것이 있다. 그것은 바로, 구멍의 개수를 계산하는 Homology 이론에 따르면 구멍의 개수에 해당하는 만큼만 Cycle이 존재하는 것이 아니라는 점이다. 예를 들어, 구멍 1개의 도넛은 1개의 Cycle을 가지는 것이 아니다. Homology 이론에 따르면, 구멍 1개의 도넛은 Cycle들을 원소로 가지는 1개의 집합을 가진다. 그리고 그 집합 안에는 여러 개의 많은 Cycle이 들어있다. 비슷하게, 만약 구멍 2개의 도넛이라면 Cycle들을 원소로 가지는 집합을 2개 가지게 된다. 이렇게 구멍의 개수는 사실 어떠한 수학적 개념의 집합 (수학에서는 Coset 이라고 한다)에 대응이 된다.

그런데, 현재 우리가 사용하고 있는 계산 기하학(Computational Geometry)의 표준 알고리즘(Standard Algorithm)은 각 구멍(hole)에 대응되는 집합에서 하나의 Cycle을 대표로 추출해서 보여준다. 도넛을 학교라고 비유해보자. 도넛에 구멍이 2개라면, 반이 2개인 학교라고 생각해보자. 즉 학교 전체에, 1반과 2반만 있는 것이다. 이때, 도넛의 구멍은 하나의 반에 해당한다. 이때 알고리즘이 집합에서 Cycle을 하나 택해서 보여주는 것은 마치 1반에서 한 명의 대표 학생을 보여주고, 2반에서도 한 명의 대표 학생을 보여주는 것과 같다. 즉 알고리즘(Algorithm)은 집합1과 집합2에서 Cycle을 하나씩 대표로 뽑아서 보여주는 것이다. 그렇다면 자연스럽게 우리는 대표로 추출되는 이 Cycle이 우리가 이용하기에 타당한지, 아니면 적어도 어떻게 그 대표 원소가 추출되는지 그 원리가 알고 싶어진다. 왜냐하면, 음악 분석에서 Cycle이 그만큼 중요하게 쓰이기 때문이다.

 기존에는 Persistent homology의 많은 계산량으로 어떠한 원리로 대표적인 Cycle을 뽑아주는지 예측할 수 없었다. 그런데 최근 우리는 Graph Theory에서 사용하는 개념인 Tree를 이용하면 어떠한 원리로 Cycle을 뽑아주는지 직관적으로 쉽게 알 수 있음을 깨달았다. 여기서 Tree란 Cycle을 하나도 가지지 않는 그래프이다.

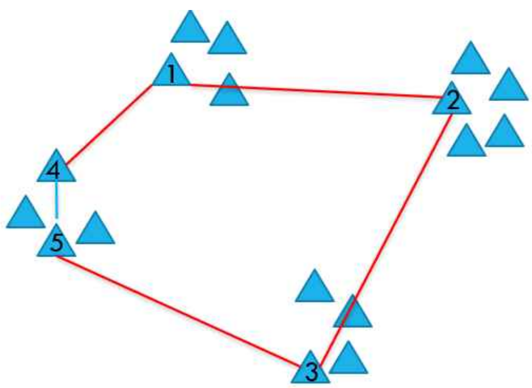


왼쪽의 그림과 같이 마치 나무처럼 생긴 모양이라 하여 Tree라 불린다. Persistent homology에서 변(edge)을 차례로 추가해 나갈 때, 트리(tree)를 아니게 만드는 변(edge)들은 무시해가며 변을 계속해서 하나씩 추가시키도록 하자. 그리고 이렇게 만들어진 트리(Tree)를 순차적 트리(Sequential Tree)라고 하자. 이 순차적 트리가 바로 Algorithm이 뽑아주는 Cycle의 구성요소를 직관적으로 보여주는 핵심개념이다. 오른쪽 그림과 만약 알고리즘이 [1, 2, 3, 4, 5]라는 대표 Cycle을 계산해 내었다고 한다면, 이는 트리를 점 1, 2, 3, 4, 5로 진행 후 마지막에 트리의 두 끝점 점 1, 5를 이어주는 모습이 되는 것이다.



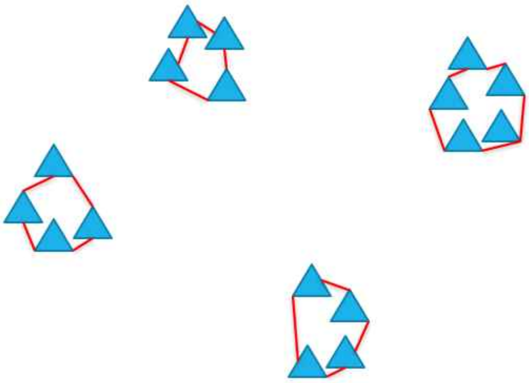

 우리는 알고리듬이 Cycle을 뽑아주는 원리가 무엇인지에 대해 알아보았다. 그런데, 순차적 트리를 만들 때, 낮은 값의 변(edge)부터 채워나가는 과정임을 생각해보자. 즉 값이 낮은 변(edge)일수록 트리(Tree)의 구성요소가 되기싫다. 우리는 이러한 발견에서 Cycle 분석방법을 대역적(Global) 방법과 국소적(Local) 방법으로 나눌 수 있음을 알게 되었다. 좀 더 자세히 알아보자.

 음악 데이터가 주어졌을 때, 어떤 두 점(Vertex)이 연속적으로 연주될 경우 그 두 점 사이의 변(edge)에 값을 1씩 늘려간다. 그러면 연속적으로 연주가 많이 되는 두 점의 경우 값이 클 것이고, 반대로 적게 연주가 되면 그 값은 작을 것이다. 예를 들어서 전체 노래에서 딱 한 번만 연속적으로 연주된 두 점이 있다고 한다면 그 값은 정확히 최솟값으로 1로 주어질 것이다. 그러면 순차적 트리는 값이 작은 변들부터 추가시키므로 트리(tree)를 구성하는 변(edge)들은 값이 낮은, 즉 연속적으로 적게 연주된 점들을 이어 만든 그래프가 될 확률이 높다. 만약, 우리가 연속적으로 많이 연주된 점들을 가까운 점들로 여기고



이를 그림으로 표현한다고 하자. 그렇다면 위의 그림과 같이, Cycle은 가깝게 모인 점들의 모임(group)들의 원소를 하나씩 고르고 전체적으로 한 바퀴 돌고 오는 모습처럼 될 것이다. 따라서 이를 우리는 대역적(Global) 분석방법이라고 한다. (참고로, 그림과 같이 뮤직네트워크를 유클리드 공간에 정확히는 넣을 수 없다. 다만, 이해를 돕기 위한 그림이다.)

그러면 이제는 국소적(Local) 방법이 무엇인지 생각해보자. 어떤 두 점(Vertex)이 연속적으로 연주될 경우 그 두 점 사이의 변(edge)에 값을 1씩 늘려가는 과정에서 그 값의 역수를 취하면 어떻게 될까? 예를 들어 5번 연주되었다면 그 역수인 1/5의 값을 부여하는 것이다. 그러면 순차적 트리(Sequential Tree)의 변(edge)들은 낮은 값부터 구성되므로 아까와는 반대로 연속 연주가 많이 된 두 점을 잇는 변들이 트리를 구성할 확률이 높다. 이를 그림으로 표현해보자면, 왼쪽 그림과 같이 가까운 점들로 Cycle을 구성하는 것처럼 된다. 따라서 우리는 이를 국소적(Local) 분석방법이라고 한다. 대역적 분석방법과 국소적 분석방법은 분석의 목적에 따라 쓰임새가 달라진다. 예를 들어, 국소적 분석방법을 통해 나온 Cycle을 이용해 기계 학습을 이용한 작곡을 한다면 연속적으로 자주 출현한 음들에 대해 작곡이 이루어질 것이다. 반대로 대역적 Cycle들을 이용한다면, 상대적으로 연속적으로 적게 연주된 음들로 노래를 구성하게 될 것이다. 또한, 대역적 분석방법으로 추출한 Cycle은 실제로 곡에 출현하진 않지만 숨겨져 있던 전체적인 곡의 구조(structure)를 발견하는 데 도움을 줄 수 있을 것이다.

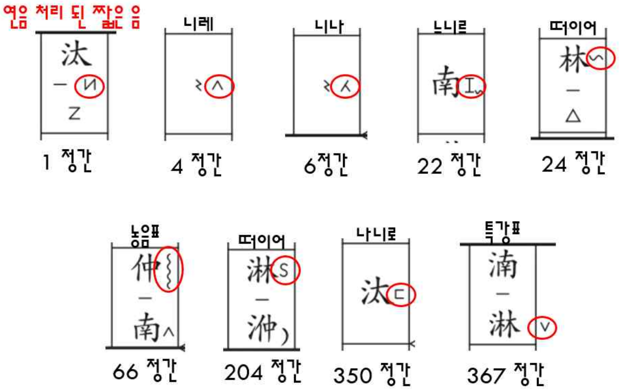


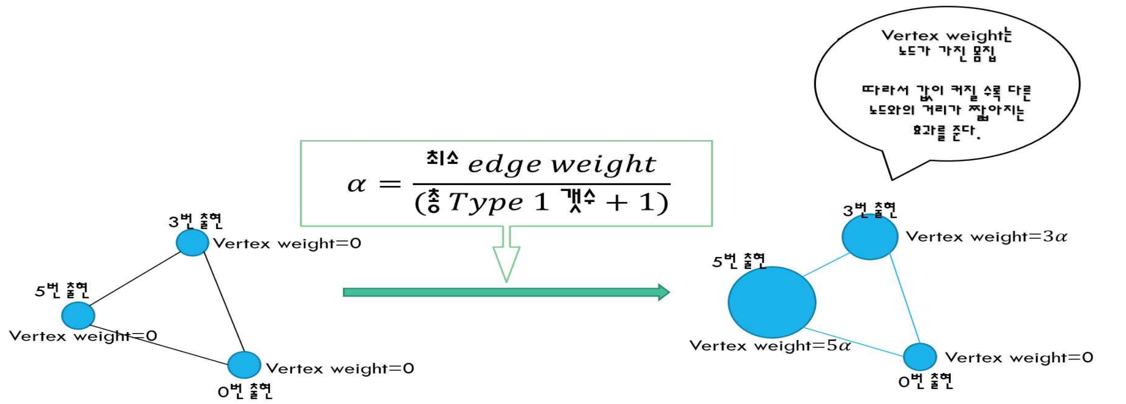
## 2. 장식음과 시김새의 음악 분석에의 적용

서양의 음악과 달리, 한국의 전통음악에서 빼놓을 수 없는 특징인 장식음과 시김새에 관해 얘기해보자. 시김새와 장식음을 우리의 분석방법에 적용하면 어떤 변화가 일어나게 될까? 장식음과 시김새를 적용하지 않고 뮤직네트워크를 그려냈을 때와 적용했을 때의 그 차이는 선명하다. 좀 더 자세히 알아보자.

우리는 뮤직네트워크를 그리기 위해 장식음과 시김새가 적용된 노드들을 2가지로 구분하였다. 데이터의 첫 번째 유형은, 노드가 주변 노드보다는 자신에게 영향을 많이 주는 것과 관련이 있는 데이터 유형이다. 대금 수연장의 예시로 보자면, 왼쪽 그림과 같이 니레, 니나 등과 같은 예시들이 있다. 이러한 데이터들은 그 음 자체에 강조를 주는 경향이 강하다. 이러한 데이터들을 Type1이라고 칭하자. 이러한 점

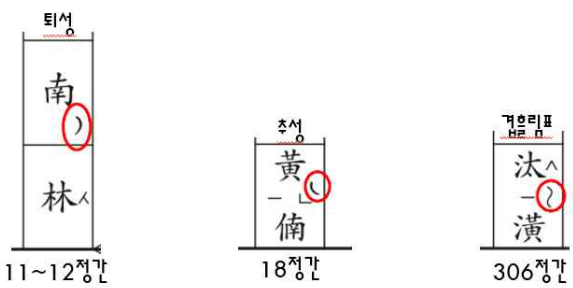
(vertex)들에 대해서는 점(vertex) 그 자체에 값(weight)을 부여해 많이 출현할수록 그 값이 커지도록 만들었다. 이러한 과정을 그림으로 표현해 보자면 아래의 그림과 같다. 값이 커질수록 해당 점들의 몸집이 커진다. 그리고 이렇게 커진 점은 다른 점과의 거리가 가까워진다.



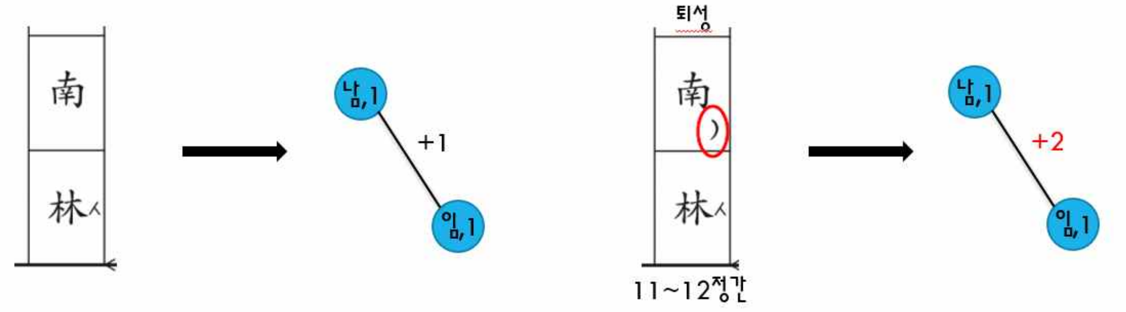




즉, 인상적인 연주의 노드일수록 다른 음과 연결되기 쉬운 점이 된다는 것이다.

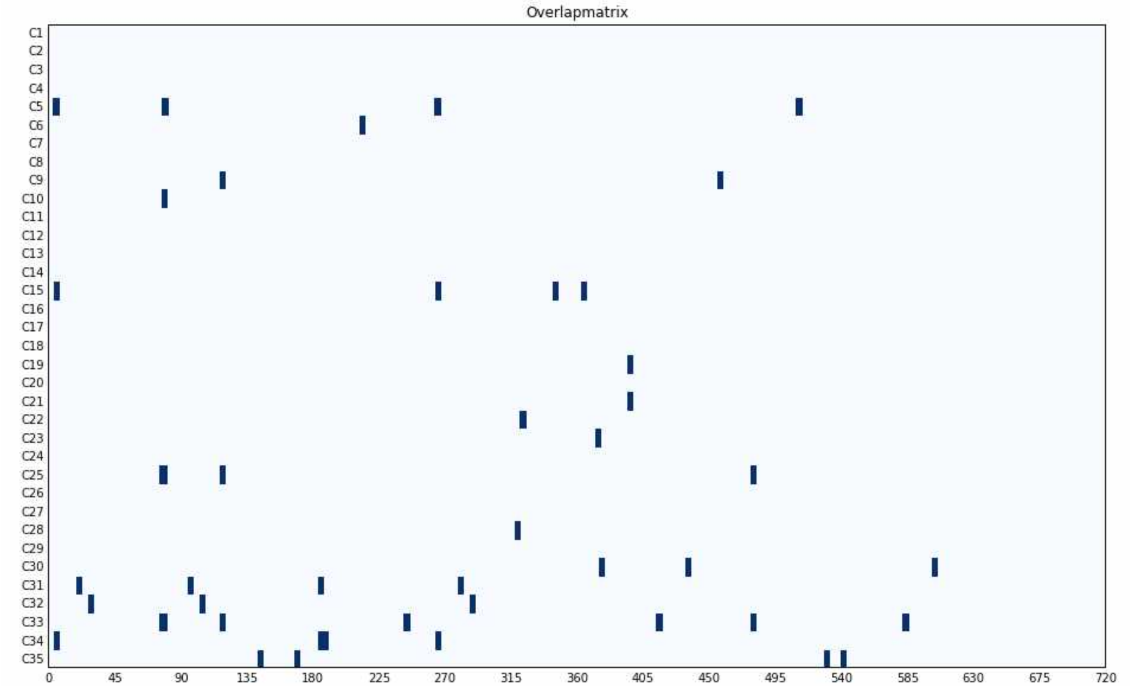
다음으로 두 번째 유형의 데이터가 있다. 대금 수연장의 예시로 보자면, 미는표(추성), 흘림표(퇴성), 겹흘림표 등이 대금정악보에 출현하고 있다. 이러한 음들은 그 자체에도 영향을 주지만 Type1의 데이터보다 다른 음과의 연결에 직접적인 영향을 준다. 이러한 데이터들을 Type2라고하자.



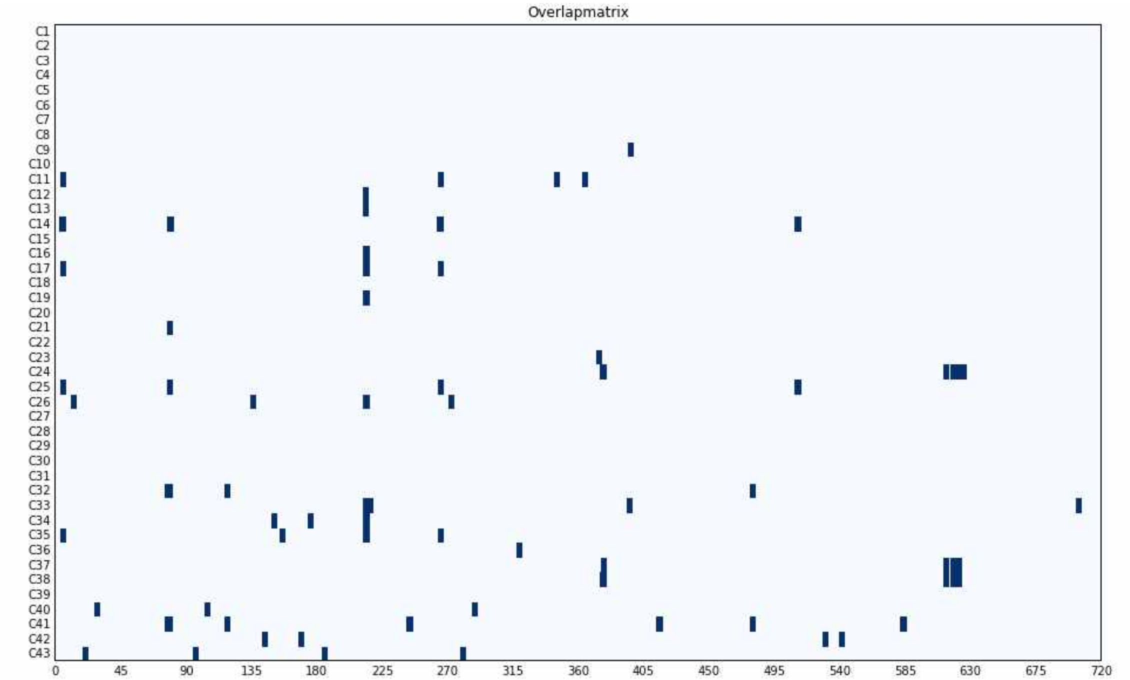

이렇게 Type2의 데이터들은 뮤직네트워크를 구성할 때, 연속적인 음에 대해 부여했던 1의 값 대신에 더 큰 값인 2를 부여하는 방법을 사용한다. 아래의 그림과 같이 퇴성이 없을 때는 값을 1을 늘렸다면, 퇴성으로 연결된 두 음은 2의 값을 추가로 부여했다. 이러한 방식으로 대금 수연장을 분석했을 때, 그 결과 차이는 어떻게



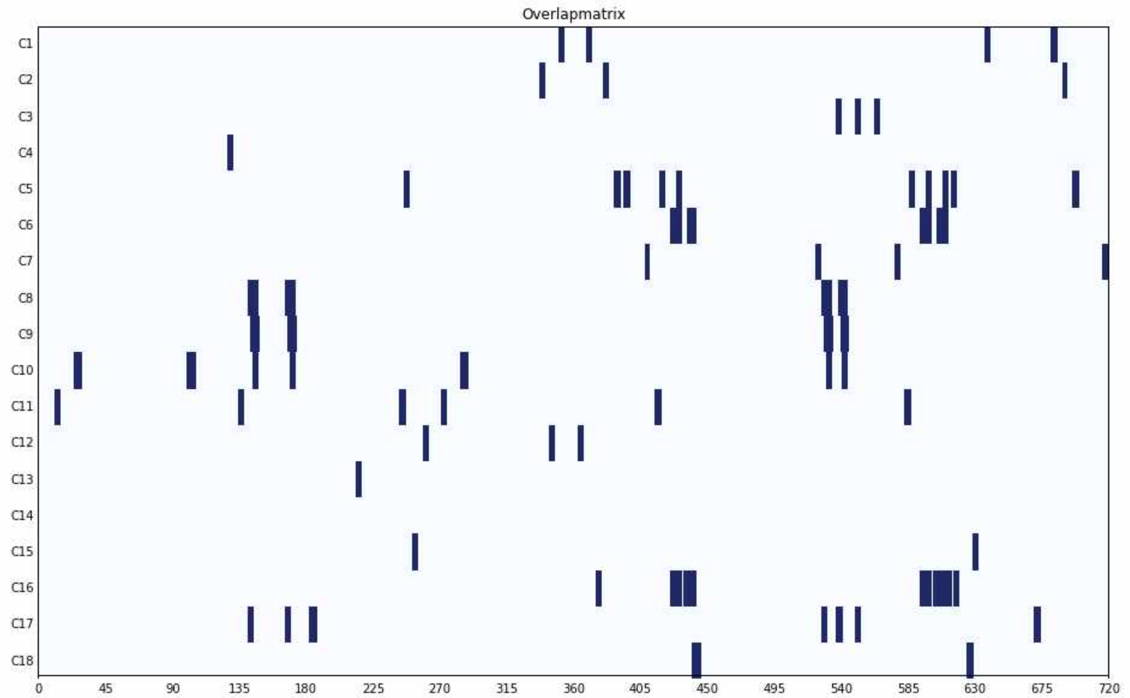
될까? 그 차이는 실제 연주에서 오는 차이만큼이나 선명하다. 대역적 분석방법부터 알아보자. 먼저 장식음과 시김새를 적용하지 않았을 때의 결과를 소개해보자면, Cycle의 추출 개수는 35개이고 Overlap Matrix는 아래 그림과 같다. 여기서 Overlap Matrix의 세로축은 추출된 Cycle들을 의미하고 가로축은 실제 연주되는 곡에서 차례대로 연주되는 점들을 의미하게 된다. 그리고 그 음들이 연속적으로 4개 이상 해당 Cycle의 점들에 속하게 될 때 색을 칠하는 규칙으로 만든 Overlap Matrix를 만든다.



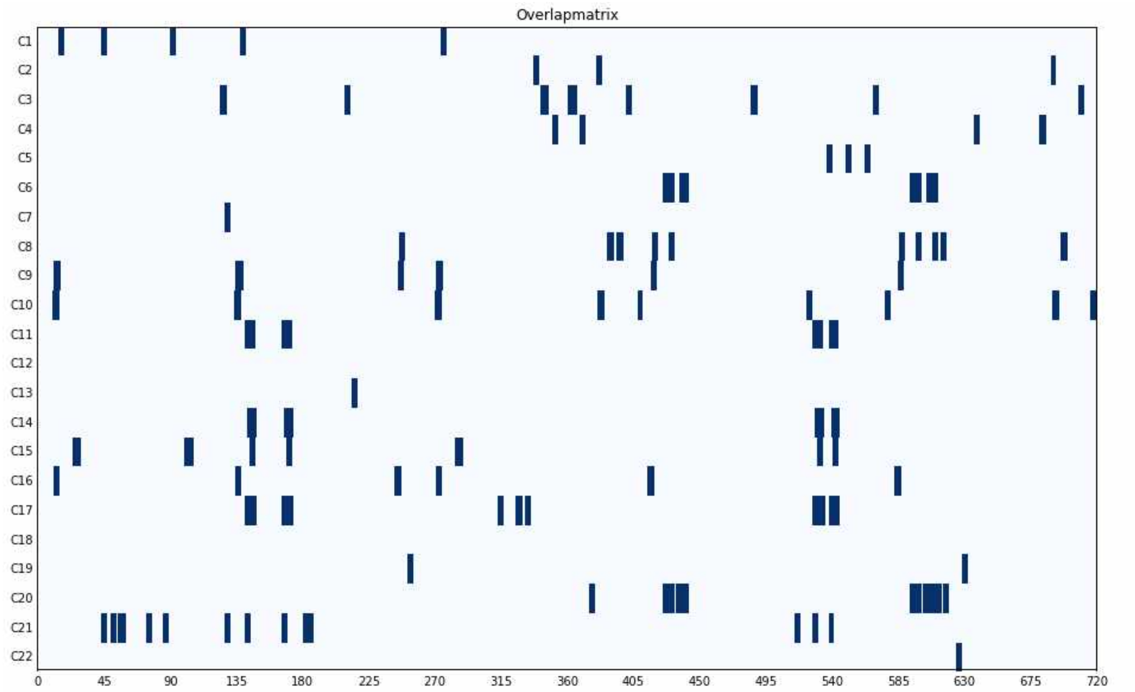

 그렇다면, 대역적 분석방법에 장식음과 시김새를 적용한 후에 어떤 변화가 생겼는지 살펴보자. 우선 대금 수연장의 Cycle의 개수는 43개가 되었다. 그리고 Overlap Matrix는 다음과 같다. Cycle의 개수가 8개가 증가하였는데도 불구하고 Overlap Matrix의 밀도는 조금 증가한 것으로 보인다. 이는 실제로 연주되는 빈도가 높아졌음을 의미한다. 하나의 결과로 결론을 내릴 수는 없겠지만, 하나의 가설은 이는 장식음과 시김새가 음의 강약이나 특색들을 더욱 뚜렷하게 구분해서 생긴 변화로 볼 수 있을 것 같다는 것이다. 장식음과 시김새를 적용하지 않았을 때는 Cycle의 개수가 35개이다. 그리고 이때 Overlap matrix를 보면 실제 노래에서의 출현도 그 빈도가 적은 것으로 관찰된다. 그러나 시김새와 장식음을 적용한 후에는 Cycle의 개수가 43개로 늘었고, 실제의 연주에서 나타나는 빈도도 확연히 높아진 것을 볼 수 있다. 이는 시김새와 장식음이 한국음악에서 굉장히 중요한 역할을 하고 있다는 사실과 비슷한 맥락으로 해석이 될 수 있겠다.




또한, 대금수연장에 대해 국소적 분석방법론을 적용한 결과에 대해 알아보자. 대역적 분석 방법에 비해, 국소적 분석 방법에서는 실제로 연속적으로 많이 연주된 음들의 변(edge)들이 순차적 트리(Tree)를 구성하게 될 것이다. 따라서, 국소적 분석으로 나온 Overlap Matrix의 장식음과 시김새의 적용 유무에 따른 변화양상을 보는 것은 또 다른 의미를 줄 수 있다. 우선 장식음과 시김새를 적용 하기 전에는 Cycle이 18개가 추출되었다. 또한, 아래의 그림과 같이 Overlap Matrix가 나왔다.



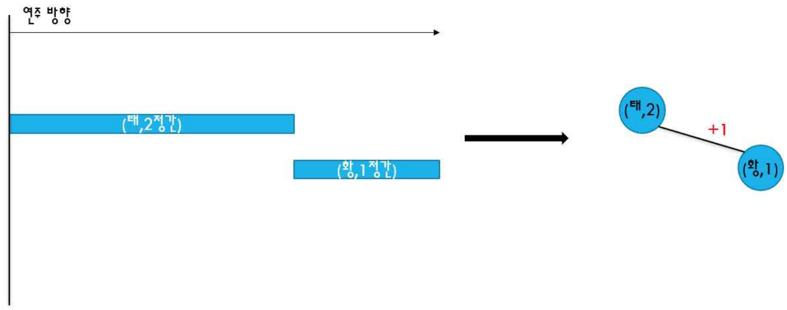
그럼 이제, 장식음과 시김새의 적용을 한 후에는 국소적 분석에 대한 결과가 어떻게 바뀌었을까? 우선 대금 수연장의 국소적 분석에 대한 Cycle의 개수는 22개이다. 또한, Overlap Matrix를 살펴보면 아래의 그림과 같다.




대역적 분석방법과 마찬가지로, 장식음과 시김새를 적용한 후에는 확연히 Overlap Matrix의 밀도가 높아진 모습이다.

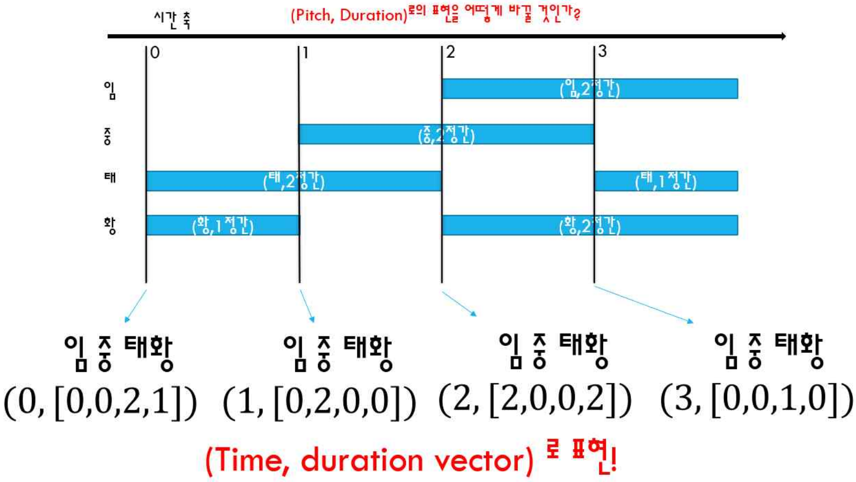
# 3. 합주 분석으로의 일반화

다음으로 합주에 대해서 알아보자. 기존의 분석 방법론에서는 하나의 단선율로 이루어진 연주에 대한 분석을 다루었다. 예를 들어, (태, 2정간) 노드의 연주 끝 시각과 (황, 1정간) 노드 연주의 시작 시각이 일치한다면 그림과 같이 두 노드 사이의 변에 값을 1 증가시키는 방식이었다. 그러나 여러 음이 동시에 연주될 수 있는 합주의 경우에는 이러한 분석방법이 어떻게 적용될 수 있을까?



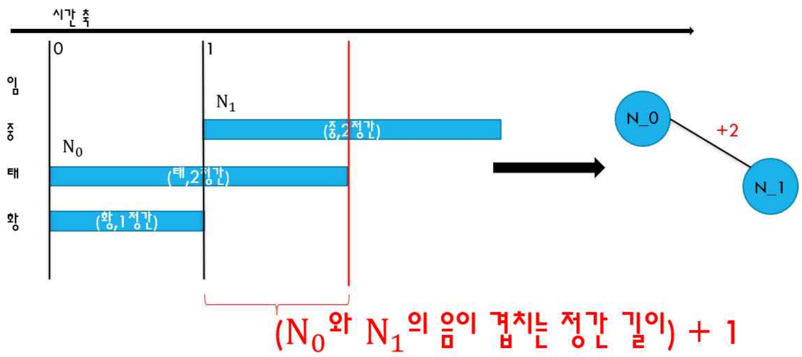

우선 첫 번째로 해결해야 할 문제는 바로 음의 표현 방식이다. 기존의 (Pitch, Duration)으로 하나의 음을 표현한 방식은 여러 Pitch와 여러 Duration이 동시에 발생

하는 음 다발에는 적합한 표현 방식이 아니다. 그런데 여기서, 음을 표현하기 위해 Pitch를 꼭 표현할 필요가 있을까 하는 의문을 제기해보고자 한다. 좀 더 자세히 말해보자면, 황이라는 음이 3정간 동안 연주된 것을 정보로 표현하기 위해서 우리는 황이라는 정보를 반드시 표시해야 할까? 결론은 아니다. '황'이라고 적지 않아도 '황'의 자리를 약속해준다면 우리는 3이라는 값만 표시하면 된다. 예를 들어, 어떤 시점에 (태, 2정간), (황, 1정간)의 음이 동시에 연주되었다고 하자. 우리는 [임, 중, 태, 황]이라는 음의 자리를 약속해주기만 한다면, [0, 0, 2, 1]로 표시하면 된다. 즉,



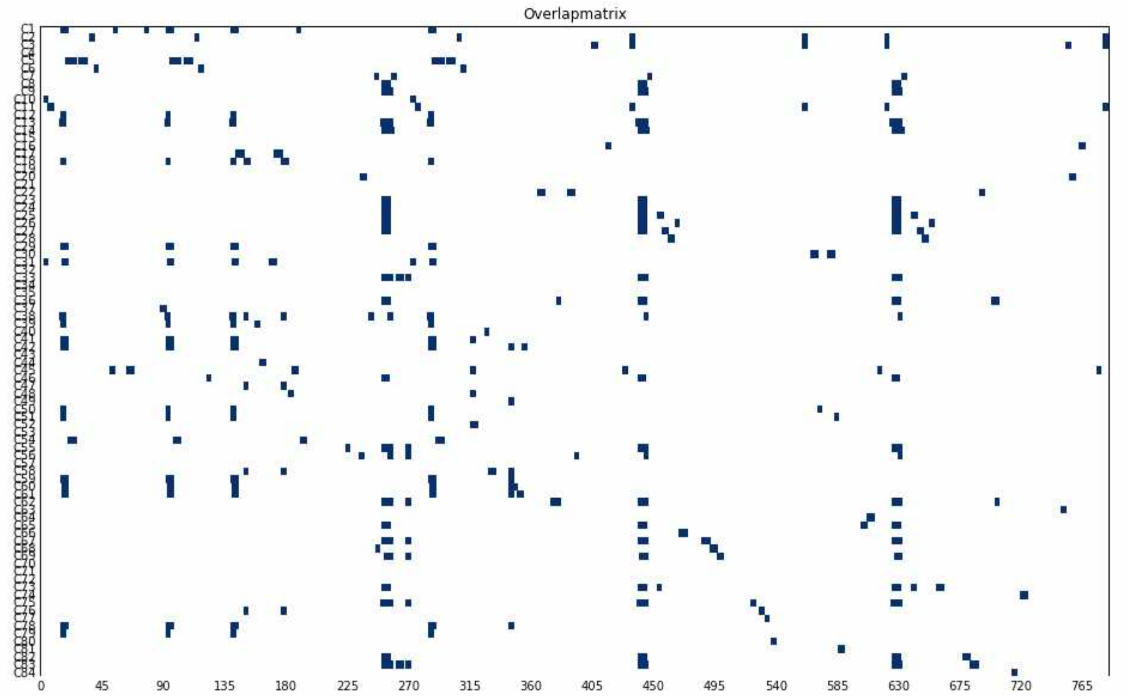

그림과 같이 노드를 (Time, Duration Vector)로 표현하면 된다. 이로써 우리는 Pitch의 표현을 없앨 수 있고, 이로써 동시에 연주되는 음 다발을 하나의 점으로 표현할 수 있는 큰 도약을 하였다.

그러면, 다음으로 해결해야 할 문제는 연속적으로 연주되는 두 점(vertex) 사이의 값을 어떻게 부여할 것인지에 대한 것이다. 기존에 단선율에서 연속적인 두 음에 대하여 부여하였던 1이라는 값과는 차별점을 두어야 할 것으로 보인다. 그것에 대한 하나의 제안으로 다음과 같은 방법이 있다. 바로 1의 값 대신에 '두 음 연주의 겹치는 정간의 길이 + 1'의 값을 주는 것이다. 이렇게 하면 기존의 단선율의 경우에는 두 음 연주의 겹치는 정간의 길이는 항상 0이므로 값이 1이 되어 기존의 단선율 정의 방식에서 벗어나지 않음을 알 수 있다. 즉, 아래의 그림과 같이 값을 부여하는 것이다. 그림을 보면, 두 음이 연주되는 정간의 길이가 1정간 만큼 겹쳤으므로 값은 2가 될 것이다. 이러한 방식으로 분석하는 방법도 역시 대역적인 분석과 국소적인 분석을 할 수 있다.




이에 대한 예시로, 수연장지곡을 대금과 가야금 2중주 연주에 대한 국소적 분석의 결과를 소개해보고자 한다. 대금과 가야금의 2중주의 경우 Cycle은 84개가 추출되

었으며, Overlap Matrix는 다음과 같다.




여기까지 우리는 위상 수학적 데이터 분석기법을 통해 한국음악을 분석하는 전반적인 과정에 대해 다루어보았다. 그리고 순차적(tree)를 이용해 알고리즘이 어떻게 Cycle을 추출해 주는지 소개하였고, 그 원리를 통해 대역적(Global) 분석과 국소적(Local) 분석이 가능함을 소개하였다. 또한, 장식음과 시김새가 뮤직네트워크 생성에 끼치는 영향을 소개함으로써 한국음악에서 장식음과 시김새가 얼마나 중요한 역할을 하는지에 대해서 간접적으로 느낄 수 있음도 소개하였다. 그리고 마지막으로 단선율뿐만 아니라 합주에 대해서도 기존의 연구방법론을 확장할 수 있음을 보였다. 그러나, 아직 해결해야 할 과제들이 많고 그만큼 발전될 가능성이 많은 연구 주제임이 분명하다.