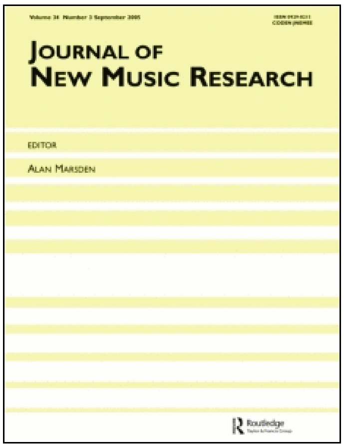


Journal of New Music Research
Publication details, including instructions for authors and subscription information:
http://www.informaworld.com/smpp/title~content=t713817838

A graph-theoretical approach for pattern matching in post-tonal music analysis

Wai Man Szeto[a]; Man Hon Wong[a]
[a] Department of Computer Science and Engineering, The Chinese University of Hong Kong, Hong Kong

To cite this Article Szeto, Wai Man and Wong, Man Hon(2006) 'A graph-theoretical approach for pattern matching in post-tonal music analysis', Journal of New Music Research, 35: 4, 307 — 321
To link to this Article: DOI: 10.1080/09298210701535749
URL: http://dx.doi.org/10.1080/09298210701535749

Journal of New Music Research
2006, Vol. 35, No. 4, pp. 307 – 321

A Graph-Theoretical Approach for Pattern Matching in Post-Tonal Music Analysis

Wai Man Szeto and Man Hon Wong

Department of Computer Science and Engineering, The Chinese University of Hong Kong, Hong Kong

Abstract

A graph-theoretical approach is proposed to facilitate pattern matching in post-tonal music analysis with pitch-class set theory, in which music perception is taken into account by incorporating stream segregation – a model of the perceptual organization of music. A piece of music is modelled as a graph, with each musical note presented as a vertex and the relationship between a pair of musical notes as an edge. The relationship is determined by stream segregation. According to the proposed matching conditions, searching for a musical pattern is equivalent to searching for a special subgraph called a *maximal matched CRP subgraph*. The comparisons are made between the patterns identified by the graph-theoretical approach and those by the musicologists.

1. Introduction

One of the most significant areas in music research, analysis of post-tonal music – art music that began in Europe in the early 20th century, significant composers of which include Bela Bartok, Alban Berg, Arnold Schoenberg, Igor Stravinsky, Anton Webern – consists of studying how this music works by finding the building blocks as unifying cells in such a musical work. Searching for these building blocks or musical patterns can be modelled as a pattern matching problem: given a musical pattern query and a music database, the aim is to search for all matched patterns in the database. In this paper, the pattern matching problem in post-tonal music analysis is based on *pitch-class set theory* – the widest currency seen as the core system for post-tonal music analysis (Bent & Pople, 2001). The idea of pitch-class set theory is illustrated in the following example in Straus (2000). The opening of Arnold Schoenberg's *Klavierstüke*, Op. 11, No. 1 is shown in Figure 1. Although the first three notes of the melody and the sustained notes in bars 4–5 look quite dissimilar, these two collections of notes are, in the perspective of pitch-class set theory, equivalent building blocks because they both contain a B, a G♯, and a G♮. Twentieth-century composers have greater freedom in exploring ways to unify a music composition and pitch-class set theory aims to uncover building blocks hidden in the music. A building block is called a *pitch-class set*, denoting a group of pitch classes. A pitch class is represented by one of the integers from 0 to 11, the naming of pitch class is based on the chromatic scale, as shown in Table 1 (Straus, 2000). Octave equivalents (such as B3 and B4) and enharmonic equivalents (such as B and C♭) are considered exact equivalents. A pitch-class set is an unordered collection of distinct pitch classes. For example, the pitches G♮, G♯, and B, represented by integers 7, 8, and 11 respectively, form a pitch-class set {7,8,11}.

Although plenty of research works of post-tonal music analysis have been done by musicologists, most of them focus on analysing and comparing a limited number of music compositions. Comparative study across numerous compositions is difficult because the identification of pitch-class sets occurred in compositions – an important aspect of pitch-class set theory – is highly time-consuming and error-prone when performed manually. Large-scale comparative study requires an effective post-tonal music analysis system.

There are software utilities to assist post-tonal music analysis (Winsor & DeLisa, 1991; Taylor, 2001; Tomlin,

Correspondence: Wai Man Szeto, Department of Computer Science and Engineering, The Chinese University of Hong Kong, Shatin, N.T., Hong Kong. E-mail: wmszeto@cse.cuhk.edu.hk

DOI: 10.1080/09298210701535749

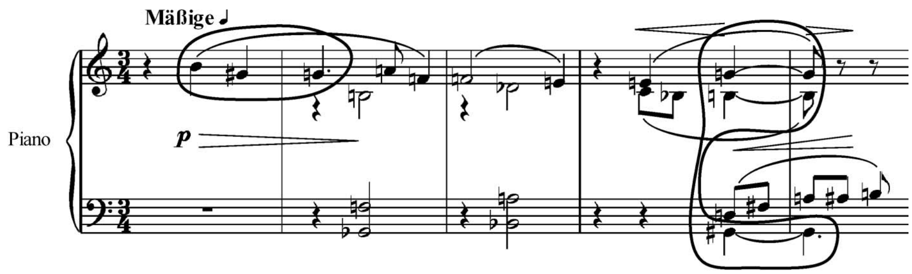


Fig. 1. The opening of Arnold Schoenberg's *Klavierstücke*, Op. 11, No. 1.

Table 1. The integer notation of pitch classes.

Integer name	Pitch-class content	Integer name	Pitch-class content
0	B♯, C, D♭♭	6	F♯, G♭
1	C♯, D♭	7	F♯♯, G, A♭♭
2	C♯♯, D, E♭♭	8	G♯, A♭
3	D♯, E♭	9	G♯♯, A, B♭♭
4	D♯♯, E, F♭	10	A♯, B♭
5	E♯, F, G♭♭	11	A♯♯, B, C♭

2003), but none of them considers searching for pitch-class sets. Forte (1966), Isaacson (1998), and Doerksen (1999) investigated how to choose groups of musical notes for analysis. Peter Castine (1994) published his PhD thesis on computer-aided analysis on post-tonal music. However, he focused on how a graphical user interface could help in pitch-class set theory analysis. A Pascal routine named *Setsearch* is presented in the book *Pascal Programming for Music Research* (Brinkman, 1990). It locates pitch-class sets by using a sequential searching algorithm with a sliding frame. However, it is difficult to determine the frame size as the author mentioned. An improper value of the frame size may cause false dismissals, i.e. some matched patterns are missed.

The fundamental problem of pattern matching in post-tonal music analysis is on the definition of matching condition(s). In Figure 1, given the pitch-class set query {7,8,11}, the first matched pattern is the first three notes of the melody but not including the minim note B♮ in bar 2. The second matched pattern is the sustained notes in bars 4–5 but not including the quaver note B♮ in bar 5. Although there is no "definite" matching condition for post-tonal music analysis, musicologists generally agree that the melodic relationship and the harmonic relationship between the musical notes in a matched pattern are significant (Williams, 1997; Straus, 2000). In the first matched pattern, the three notes are the first three notes of the melody. In the second matched pattern, the three notes are sounding simultaneously and they are related harmonically.

The melodic relationship can be described in the *stream segregation* model. When listening to music, people perceive music in groupings of musical notes[1] called *streams* which are the perceptual impression of a connected series of musical notes, instead of isolated sounds (Bregman, 1990). The process of grouping musical notes into streams is called *stream segregation*[2]. A melody is usually perceived as a single coherent and continuous musical line, that is, a stream. In Szeto & Wong (2006), we present a stream segregation algorithm to identify streams in music compositions which are represented symbolically in MIDI files. In order to consider the melodic relationship, we propose the graph-theoretical approach in which stream segregation is incorporated into the matching conditions. A piece of music is modelled as a graph. Each musical note is presented as a vertex and the relationship between a pair of musical notes as an edge. The pairwise relationship is determined by stream segregation. Searching for a pitch-class set is equivalent to searching for a special subgraph called *maximal matched complete r-partite path subgraph*. A widely studied piece – Schoenberg's *Klavierstücke*, Op. 11, No. 1 – will be analysed by the graph-theoretical approach as an illustrating example. Comparisons are also made between the pitch-class sets identified by the graph-theoretical approach and those by the musicologists.

This paper focuses on the computational aspect of post-tonal music analysis with pitch-class set theory. In addition to Schoenberg's *Klavierstücke*, Op. 11, No. 1, more music compositions can be analysed by the graph-theoretical approach to study how composers unify a

[1]The terms *pitch* and *musical note* are sometimes confusing. In this paper, a pitch is a musical sound with certain fundamental frequency; a musical note is a musical sound of a given pitch and duration.

[2]In Bregman (1990), *stream segregation* is divided into two processes: simultaneous integration and sequential integration. Simultaneous integration is the process of grouping frequencies into musical notes. Sequential integration is the process of grouping musical notes into streams. However, in this paper, *stream segregation* refers to the latter process only.

music composition with hidden building blocks. Applications of the approach include identification of dominating pitch-class sets in a composition and comparative study across numerous compositions of the same or different composers, shedding light on the following questions: how are pitch-class sets used in a work? Are there idiosyncrasies in their usage with a particular composer? Are there any sets that can be viewed as "common wealth" among post-tonal composers? How such music works? The graph-theoretical approach is likely to be a valuable tool for all these questions.

The following section briefly explains the stream segregation model in Szeto & Wong (2006). Given a music composition represented symbolically, the stream segregation algorithm identified all of its streams. This information is the foundation of representing music as a graph which is discussed in Section 3. According to the matching conditions proposed in Section 4, the algorithm presented in Section 5 finds all the subgraphs matched the given pitch-class set query in the graph representing a music composition. The results of the analysis of Schoenberg's *Klavierstücke*, Op. 11, No. 1, are shown in Section 6. A conclusion is given in Section 7.

2. Stream segregation

A stream is the perceptual impression of a connected series of musical notes; stream segregation is the process of grouping musical notes into streams. In Szeto & Wong (2006), stream segregation is modelled as a clustering process and an adapted single-link clustering algorithm is proposed. A stream is represented as a cluster[3]. In the stream segregation model, each musical note is represented as an *event*. An event $\mathbf{e}$ is a vector (t^s, t^e, p), where t^s is the start time, t^e is the end time, and p is the pitch. There are two kinds of the relation between two events: *sequential events* and *simultaneous events*. Given two events, if their durations overlap each other, they are simultaneous events. Otherwise, they are sequential events. The definitions are formally given below:

Definition 1 (Sequential events). *Given two events* $\mathbf{e}_i = (t_i^s, t_i^e, p_i)$ *and* $\mathbf{e}_j = (t_j^s, t_j^e, p_j)$, *they are* sequential events *if* $t_i^e \le t_j^s$ *or* $t_j^e \le t_i^s$.

Definition 2 (Simultaneous events). *If two events* $\mathbf{e}_i$ *and* $\mathbf{e}_j$ *are not sequential events, they are said to be* simultaneous events.

These definitions are further illustrated by the following five-event example:

Example 1. *In Figure 2(a), the event pairs* $\{\mathbf{e}_1, \mathbf{e}_4\}$, $\{\mathbf{e}_1, \mathbf{e}_5\}$, $\{\mathbf{e}_2, \mathbf{e}_3\}$, $\{\mathbf{e}_2, \mathbf{e}_4\}$, $\{\mathbf{e}_2, \mathbf{e}_5\}$, *and* $\{\mathbf{e}_3, \mathbf{e}_5\}$ *are sequential events because they do not overlap each other in time. The event pairs* $\{\mathbf{e}_1, \mathbf{e}_2\}$, $\{\mathbf{e}_1, \mathbf{e}_3\}$, $\{\mathbf{e}_3, \mathbf{e}_4\}$, *and* $\{\mathbf{e}_4, \mathbf{e}_5\}$ *are simultaneous events.*

The objective of the adapted single-link clustering algorithm is to connect an event to its "nearest" sequential event. The "distance" between two events is defined by the *inter-event distance* in the pitch–time space. The start time and the end time of an event are in the unit of seconds. The pitch of an event is transformed to the frequency in the *mel scale* (O'Shaughnessy, 1987). Given two events $\mathbf{e}_i = \left(t_i^s, t_i^e, p_i\right)$ and $\mathbf{e}_j = \left(t_j^s, t_j^e, p_j\right)$, and $t_i^e \le t_j^s$, the *inter-event distance EDIST* is

$$EDIST(e_i, e_j) = \begin{cases} \sqrt{\left(\alpha\left(t_i^e - t_j^s\right)\right)^2 + (f_i - f_j)^2} & \\ \quad \text{if } \mathbf{e}_i \text{ and } \mathbf{e}_j \text{ are sequential events,} & \\ \infty \quad \text{otherwise,} & \end{cases}$$

where f_i and f_j are the mel-scale frequency of the pitches p_i and p_j respectively, and α is the weighting factor determined empirically to be 1000 in Szeto & Wong (2006).

In the clustering algorithm, the inter-cluster distance between two clusters is defined as below. If all pairs of events drawn from one cluster and another cluster are sequential events, the inter-cluster distance is the minimum of the inter-event distances among all such pairs of events; otherwise, the inter-cluster distance is set to infinity to ensure that no pair of simultaneous events is in the same cluster. The clustering algorithm starts with the clustering that each event itself forms a cluster. Among all possible pairs of clusters, the algorithm finds the pair which has the minimum inter-cluster distance over all other pairs. Then the pair of clusters is merged into a larger cluster. Thus, a new clustering is formed and the process is repeated. This forms clusters or *streams* which are chains of sequential events. A stream identified by the algorithm is a set of events, in which every pair of events is sequential events. Every event belongs to one and only one stream. The details of the algorithm can be found in Szeto & Wong (2006). The output of the algorithm for the five-event example is presented in Figure 2(a), where $\{\mathbf{e}_1, \mathbf{e}_4\}$ is a stream and $\{\mathbf{e}_2, \mathbf{e}_3, \mathbf{e}_5\}$ is another. The next section will discuss how the output streams are used to represent music by graph theory.

3. Music representation

In order to formulate the matching conditions of pitch-class set theory, music is represented as undirected graphs called *musical graphs*. The notation and the summary of graph theory are included in the Appendix. Each musical note is a vertex and the relationship between a pair of musical notes is an edge. To put it in

[3]In this paper, the term *cluster*, a term used in the study of data mining, refers to a group of similar objects.

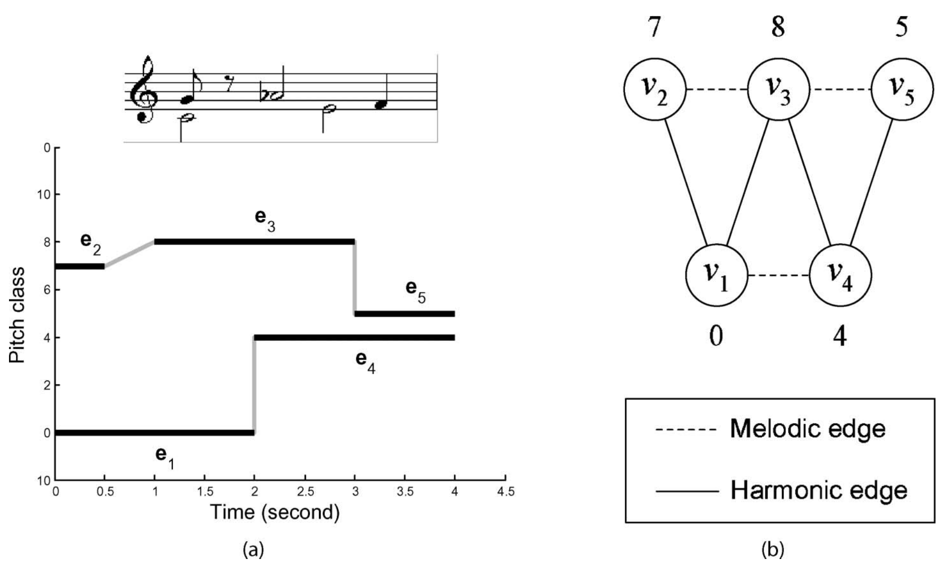


Fig. 2. The five-event example. (a) The streams identified by the stream segregation algorithm: $\{\mathbf{e}_1, \mathbf{e}_4\}$ is a stream and $\{\mathbf{e}_2, \mathbf{e}_3, \mathbf{e}_5\}$ is another stream. (b) The corresponding musical graph. Vertex v_i corresponds to the event $\mathbf{e}_i$. The number next to a vertex is the pitch class of the vertex.

the stream segregation model, an event is a vertex and a pairwise relationship of events is an edge. There are two kinds of edges: *melodic edge* and *harmonic edge*. Two adjacent events in the same stream are joined by a *melodic edge* so melodic edges can be identified by the stream segregation algorithm. For the stream $\{\mathbf{e}_2, \mathbf{e}_3, \mathbf{e}_5\}$ in Figure 2(a), the events $\mathbf{e}_2$ and $\mathbf{e}_3$ are connected by a melodic edge while the events $\mathbf{e}_3$ and $\mathbf{e}_5$ are connected by another melodic edge. Two simultaneous events are joined by a *harmonic edge*. An example in Figure 2(a) is that a harmonic edge connects the events $\mathbf{e}_1$ and $\mathbf{e}_2$. Harmonic edges can be easily identified by checking any two events that their durations overlap each other.

Given a musical graph $G = (V, E)$, the set of vertices is V; the set of edges is E; the set of melodic edges is E_m; the set of harmonic edges is E_h; and $E = E_m \cup E_h$. If two vertices v and w are joined by a melodic edge, we say that v is *melodically adjacent* to w, or vice versa. If two vertices v and w are joined by a harmonic edge, we say that v is *harmonically adjacent* to w, or vice versa. Each vertex has a label to represent its pitch class. The label of a vertex v is denoted by $label(v)$. A pitch class is an element in U where $U = \{0,1,\ldots,11\}$. Thus, $label(v)$ is in U. The musical graph representation can be further illustrated by the following example.

Example 2. *The musical graph of the five-event example is shown in Figure 2(b). A vertex v_i corresponds to the event $\mathbf{e}_i$. Dotted lines represent melodic edges; while solid lines denote harmonic edges. The number next to a vertex is the pitch class of the vertex.*

The set of all vertices melodically adjacent to a vertex v is denoted by $Adj_m(v)$. The set of all vertices harmonically adjacent to a vertex v is denoted by $Adj_h(v)$. Two events cannot be both sequential and simultaneous. Therefore, a vertex cannot be both melodically adjacent and harmonically adjacent to another vertex, thus $Adj_m(v) \cap Adj_h(v) = \phi$.

A *melodic path* $J = (P, E)$ is a path in which all edges are melodic such that $E = E_m$. Hence, a stream can be defined as a melodic path. For notational convenience, a single isolated vertex is also defined as a melodic path. The output of the stream segregation algorithm can be redefined as a set of vertex sets of melodic paths, denoted by $P = \{P_1, P_2, \ldots, P_r\}$ where r is the number of streams. As melodic edges only join vertices in the same stream, there does not exist any melodic edge between vertices in different P_i. Thus, each P_i is a *maximal melodic path* such that adding any vertex not in P_i cannot form a melodic path. An r-partite musical graph contains r streams or r maximal melodic paths. For instance, there are two maximal melodic paths $P_1 = \{v_1, v_4\}$ and $P_2 = \{v_1, v_3, v_5\}$ in Figure 2(b) of Example 2. The subgraphs $\{v_2, v_3\}$ and $\{v_3, v_5\}$ are melodic paths but they are not maximal.

According to the stream segregation model, each event belongs to exactly one stream. Therefore, each vertex belongs to one and only one maximal melodic path such that given a graph $G(V,E)$ and the stream segregation output $P = \{P_1, P_2, \ldots, P_r\}$, it is obvious that $V = P_1 \cup P_2 \cup \cdots \cup P_r$ and $P_i \cap P_j = \phi$ for $i \neq j$. In other words, if there is an edge between two vertices in different maximal melodic paths, the edge must be a harmonic edge. We denote the vertex set of the maximal melodic path containing the vertex v by $path(v)$. If $v \in P_i$, then $P_i = path(v)$. Two vertices are in the same maximal melodic path if and only if there exists a melodic path between them. If two vertices v and w are in the same maximal melodic path, it is denoted by $v \sim w$; otherwise,

$v \nsim w$. In Figure 2(b) of Example 2, the maximal melodic path $P_1 = \{v_1, v_4\}$ gives $v_1 \sim v_4$ while $P_2 = \{v_2, v_3, v_5\}$ gives $v_2 \sim v_3$, $v_3 \sim v_5$ and $v_2 \sim v_5$. The vertices not in the same maximal melodic path are $v_1 \nsim v_2$, $v_1 \nsim v_3$, $v_1 \nsim v_5$, $v_2 \nsim v_4$, $v_3 \nsim v_4$, and $v_4 \nsim v_5$.

In short, given a musical graph G, after removing all harmonic edges from G, it consists of maximal melodic paths; while after removing all melodic edges from G, it becomes a harmonic-edged r-partite graph and all vertices in the same stream form a partition class.

4. Matching conditions

Although there is no "definite" matching condition for post-tonal music analysis, musicologists generally agree that the melodic relationship and the harmonic relationship between the musical notes in a matched pattern are significant (Williams, 1997; Straus, 2000). These two relationships are captured in the stream segregation model: two musical notes related melodically are equivalent to two adjacent events in the same stream; while two musical notes related harmonically are equivalent to two simultaneous events. To preserve both melodic and harmonic relationships between musical notes in a matched pattern or an *answer*, we propose that given a musical graph and a pitch-class set query, an answer is a complete harmonic-edged r-partite subgraph, in which the vertices in each partition class form a melodic path, and the set of pitch classes in the subgraph equal to the pitch-class set query. This implies that given any two notes in an answer, they are either (i) connected by harmonic edge or melodic edge, or (ii) in the same melodic path connected by other notes in the answer. To state formally, we need the following definitions.

Definition 3 (Complete r-partite path graph). *A graph $G = (V, E)$ is a complete* r-partite path graph (CRP graph) *where $r \geq 1$ if for any two vertices v and w in V, if $v \nsim w$, then $vw \in E_h$; if $v \sim w$, then there exists a melodic path $J = (P, F)$ such that $J \subseteq G$, $F \subseteq E_m$ and v, $w \in P$.*

This can be interpreted that by removing all melodic edges from a CRP graph, it becomes a harmonic-edged complete r-partite graph; while removing all harmonic edges from a CRP graph, it consists of melodic paths.

Definition 4 (Maximal CRP subgraph). *A CRP subgraph G' of G is maximal if there does not exist a vertex v in $G - G'$ such that $G' + v$ is a CRP graph.*

This means that a larger CRP subgraph cannot be obtained by adding a vertex to a maximal CRP subgraph. Maximal CRP subgraphs are illustrated in the example below.

Example 3. *A hypothetical musical graph is shown in Figure 3. In this musical graph, there are three maximal CRP graphs constructing from the following vertex sets: $V_1 = \{v_1, v_2, v_3, v_6, v_7, v_8, v_{10}, v_{11},\}$; $V_2 = \{v_4, v_5, v_6, v_7, v_8\}$; and $V_3 = \{v_4, v_9\}$.*

Definition 5 (Matched CRP subgraph). *Given a pitch-class set Q, a CRP subgraph $G' = (V', E')$ of G is matched to Q if $Q = \bigcup_{v \in V'} label(v)$.*

In other words, the set of pitch classes in a matched CRP subgraph is equal to the pitch-class set query. An example is shown as follows.

Example 4. *Given the musical graph in Figure 3 and the pitch-class set $Q = \{0,1\}$, there are many possible matched CRP subgraphs to Q. Some of their vertex sets are listed below: $\{v_1, v_2\}$, $\{v_4, v_5, v_6\}$, $\{v_4, v_6, v_7, v_8\}$, $\{v_8, v_{10}\}$, and etc.*

After explaining the above definitions, we state the formal definition of the matching conditions below:

> *Given a musical graph G and a pitch-class set query Q, a subgraph G' is an answer to Q if and only if G' is a maximal matched CRP subgraph to Q.*

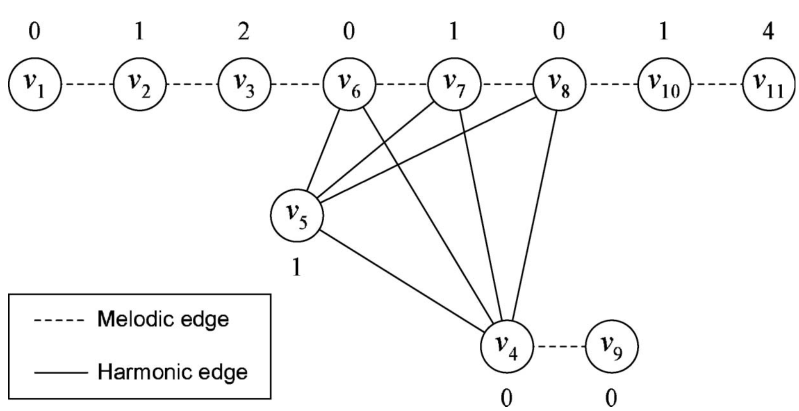


Fig. 3. A musical graph.

To put the matching conditions in another way, suppose that a subgraph H of G contains only the set of all vertices with pitch classes in the pitch-class set Q. A maximal matched CRP subgraph G' to a pitch-class set Q is that G' is matched to Q and G' is a maximal CRP subgraph of H. An example is given below to illustrate the matching conditions.

Example 5. *Given the musical graph in Figure 3 and the pitch-class set query* $Q=\{0,1\}$, *there are three answers:* $V_1=\{v_1, v_2\}$; $V_2=\{v_4, v_5, v_6, v_7, v_8\}$; *and* $V_3=\{v_6, v_7, v_8, v_{10}\}$.

In the perspective of music, the answers V_2 and V_3 are strongly related because they share some vertices with each other. However, they are listed separately. To reduce the number of answers for musicologists to interpret the result more easily, if any two answers in an answer set share one or more vertices, these two answers are taken as union to form a new answer. This procedure is repeated on an answer set until no answers share the same vertex. In Example 5, there are two answers after merging: $V_1=\{v_1, v_2\}$ and $V_2=\{v_4, v_5, v_6, v_7, v_8, v_{10}\}$.

To sum up, the graph-theoretical approach to the post-tonal music analysis problem involves the following two procedures.

1. Given a musical graph and a pitch-class set query, find all maximal matched CRP subgraphs.
2. Merge these subgraphs until no two different new subgraphs share the same vertex.

In the next section, we will discuss how to solve the problem according to the above procedures by a backtracking algorithm.

5. Algorithm

Finding all maximal matched CRP graphs is a combinatorial search problem. Given a musical graph G and a pitch-class set query Q, we solve the problems of finding all maximal matched subgraphs and merging these subgraphs by the following three steps.

1. In G, remove all vertices of the pitch classes in $U-Q$ where $U=\{0,1,\ldots,11\}$, i.e. only the vertices with the pitch classes in Q are retained. The new reduced graph is $G^*=(V^*, E^*)$.
2. Find all matched CRP subgraphs to Q in G^*.
3. Merge these subgraphs until no two different new subgraphs share the same vertex.

Noted that the matched subgraphs generated by Step 2 may not be maximal but merging all matched subgraphs gives the same result as merging all maximal matched subgraphs. This is because all possible matched subgraphs in each maximal matched subgraph are found in Step 2. Steps 1 and 3 are straightforward while Step 2, finding all matched CRP subgraphs, is difficult. This problem is closely related to enumeration of all cliques. Indeed, a clique is a special case of CRP graphs. If an n-vertex CRP graph has n partition classes, i.e. each partition class has only one vertex, this CRP graph is a clique in which all edges are harmonic edges. In order to find all matched subgraphs, we first find all CRP subgraphs and then for each CRP subgraph, we check whether it is matched to the pitch class set query. We extend a clique enumerating algorithm by Kreher and Stinson (1999) to find all CRP subgraphs. The Kreher–Stinson algorithm, or the KS-algorithm, works recursively to find all cliques without repetition by backtracking. Our algorithm is based on the KS-algorithm and adopts its notation.

To generate all of the CRP subgraphs of the reduced graph $G^*=(V^*, E^*)$ by backtracking, we need to define a partial solution and propose a scheme to compute the choice set C_t. A partial solution is a sequence $X=\langle x_1, x_2,\ldots,x_t\rangle$ of vertices such that $\{x_1, x_2,\ldots,x_t\}$ is a CRP graph. We write $X_{1:t}=\langle x_1,x_2,\ldots,x_t\rangle$.

The choice set C_t contains vertices which can be used for the extension of $X_{1:t}$. Adding any one vertex in C_t to $X_{1:t}$ gives a CRP graph. The choice set C_t can be computed by the harmonic choice set C_t^{h} and the melodic choice set C_t^{m}. Any vertex in a harmonic choice set C_t^{h} connects all vertices in $X_{1:t-1}$ by harmonic edges. The harmonic choice set C_t^{h} is defined as follows:

$$C_t^{\mathrm{h}} = \left\{v \in (V^* - X_{1:t-1}) \mid vx \in E_{\mathrm{h}}^* \text{ for each } x \in X_{1:t-1}\right\}.$$

Then the harmonic choice set C_t^{h} can be expressed in terms of C_{t-1}^{h}:

$$C_t^{\mathrm{h}} = \left\{v \in (C_{t-1}^{\mathrm{h}} - \{x_{t-1}\}) \mid vx_{t-1} \in E_{\mathrm{h}}^*\right\}.$$

The definition of the melodic choice set needs more explanation. The output of stream segregation is a set of melodic paths $P=\{P_1, P_2,\ldots,P_r\}$. The partial solution $X_{1:t-1}$ is a CRP subgraph so $X_{1:t-1}$ contains some melodic subpaths of the melodic paths in P. These subpaths are denoted by $\{P'_{i_1}, P'_{i_2},\ldots,P'_{i_k}\}$ such that $P'_{i_x} \subseteq P_i$ for some i, $X_{1:t-1} = P'_{i_1} \cup P'_{i_2} \cup \cdots \cup P'_{i_k}$, and $P'_{i_x} \cap P'_{i_y} = \phi$ for $i_x \neq i_y$. Then a melodic choice set C_t^{m} is a vertex set such that any vertex in C_t^{m} is melodically adjacent to an end of a subpath P'_{i_x} and the vertex is harmonically adjacent to all vertices in $(X_{1:t-1}-path(v))$. The vertex set of all vertices melodically adjacent to a subpath end of $X_{1:t-1}$ but not in $X_{1:t-1}$ is written as the auxiliary set $pathAdj(X_{1:t-1})$:

$$pathAdj(X_{1:t-1}) = \left\{v \in (V^* - X_{1:t-1}) \mid vx \in E_{\mathrm{m}}^* \text{ for some } x \in X_{1:t-1}\right\}.$$

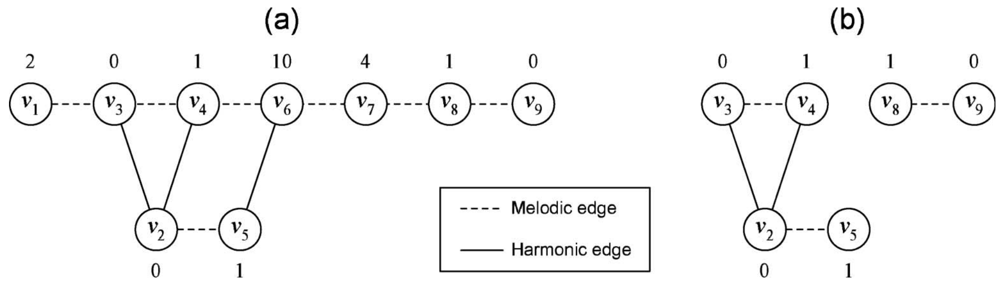


Fig. 4. (a) The musical graph G for Example 6. (b) Its reduced graph G^* given the pitch-class set query $Q = \{0,1\}$.

It can be further illustrated by the musical graph in Figure 3. If $X_{1:2} = \langle v_4, v_7 \rangle$, then $pathAdj(X_{1:2}) = \{v_6, v_8, v_9\}$.

We define the melodic choice set C_t^{m} as follows:

$$C_t^{\mathrm{m}} = \{ v \in pathAdj(X_{1:t-1}) \mid vx \in E_{\mathrm{h}}^* \text{ for each } x \in (X_{1:t-1} - path(x)) \}.$$

In the computation of C_t^{m}, all vertices in $pathAdj(X_{1:t-1})$ are checked whether each of them is harmonically adjacent to all vertices in $(X_{1:t-1} - path(x))$. If a vertex v is in $pathAdj(X_{1:t-1})$ but it is not harmonically adjacent to all vertices in $(X_{1:t-1} - path(x))$, it is rejected in the computation of C_t^{m}. In the computation of C_{t+1}^{m}, this rejected vertex is still in $pathAdj(X_{1:t})$ so the computation of C_{t+1}^{m} involves v again. In order to avoid such duplicated computation, a rejection set sequence $\mathcal{D} = \langle D_1, D_2, \ldots, D_t \rangle$ is used to store the rejected vertices. The vertex set D_t stores the rejected vertices in the computation of C_t^{m}. $\mathcal{D}_{1:t}$ denotes the vertex set $D_1 \cup D_2 \cup \cdots \cup D_t$. We define D_t below:

$$D_t = \{ v \in pathAdj(X_{1:t-1}) \mid vx \notin E_{\mathrm{h}}^* \text{ for some } x \in (X_{1:t-1} - path(x)) \}.$$

Then, we redefine the melodic choice set C_t^{m}:

$$C_t^{\mathrm{m}} = \{ v \in (pathAdj(X_{1:t-1}) - D_{t-1}) \mid vx \in E_{\mathrm{h}}^* \text{ for each } x \in (X_{1:t-1} - path(x)) \}.$$

Now, the choice set C_t is the union of the harmonic choice set C_t^{h} and the melodic choice set C_t^{m}:

$$C_t = C_t^{\mathrm{h}} \cup C_t^{\mathrm{m}}.$$

The initial values of the sets are $C_1 = C_1^{\mathrm{h}} = V^*$ and $C_1^{\mathrm{m}} = D_1 = \phi$. Finding CRP subgraphs by using the choice sets is illustrated in Example 6.

Example 6. *Given the pitch-class set query $Q = \{0, 1\}$, a musical graph G and its reduced graph G^* are depicted in Figure 4. The choice sets and the partial solutions are shown in Table 2. Note that x_t in X is chosen arbitrarily from the choice set C_t.*

Table 2. Choice sets and partial solutions in Example 6. A vertex x_t in X is chosen arbitrarily from the choice set C_t.

t	C_t^{m}	C_t^{h}	C_t	X
1	ϕ	$\{v_2, v_3, v_4, v_5, v_8, v_9\}$	$\{v_2, v_3, v_4, v_5, v_8, v_9\}$	$\langle v_2 \rangle$
2	$\{v_5\}$	$\{v_3, v_4\}$	$\{v_3, v_4, v_5\}$	$\langle v_2, v_4 \rangle$
3	$\{v_3\}$	ϕ	$\{v_3\}$	$\langle v_2, v_4, v_3 \rangle$
4	ϕ	ϕ	ϕ	$\langle v_2, v_4, v_3 \rangle$

However, an algorithm based on the above choice function will generate duplicated CRP subgraphs. In Example 6, in addition to $\langle v_2, v_4, v_3 \rangle$, $\langle v_2, v_3, v_4 \rangle$, $\langle v_3, v_2, v_4 \rangle$, $\langle v_3, v_4, v_2 \rangle$, $\langle v_4, v_2, v_3 \rangle$, and $\langle v_4, v_3, v_2 \rangle$ are also generated by such an algorithm. In the KS-algorithm of enumerating all cliques, the repetitions are avoided by arbitrarily placing a total ordering "$<$" on the vertices V. Then the vertex set V becomes a sequence, i.e. $V = \langle v_1, v_2, \ldots, v_n \rangle$, where $v_1 < v_2 < \cdots < v_n$. Only the vertices greater than x_{t-1} are put in the choice set C_t. Hence, in Example 6, only $\langle v_2, v_3, v_4 \rangle$ is generated if the total ordering is $v_1 < v_2 < \cdots < v_9$.

Nevertheless, unlike enumerating all cliques, a total ordering cannot be assigned arbitrarily to the vertices of a musical graph. After a total ordering is assigned to a musical graph, x_i is greater than x_{i-1} in the partial solution X. This means that a CRP subgraph can only be discovered by repetitively visiting a smaller vertex to a greater vertex. Then the graph becomes a digraph such that all edges are directed from a smaller end to a greater end. Hence, a directed Hamiltonian path must exist from the smallest vertex to the greatest vertex in a CRP subgraph in order to discover it.

A directed Hamiltonian path must exist for the smallest vertex and the largest vertex in a clique for arbitrary total ordering. However, it is not the case in CRP subgraphs. Consider a CRP graph with two different total ordering assignments in Figure 5. In Figure 5(a), a directed Hamiltonian path exists from v_1 to v_4, but in Figure 5(b), no such path exists so this CRP subgraph is missed. In order to guarantee the existence of a directed Hamiltonian path, we propose the total ordering of vertices as follows:

Theorem 1. *The vertices are sorted by the start time of their corresponding events in ascending order. If there are*

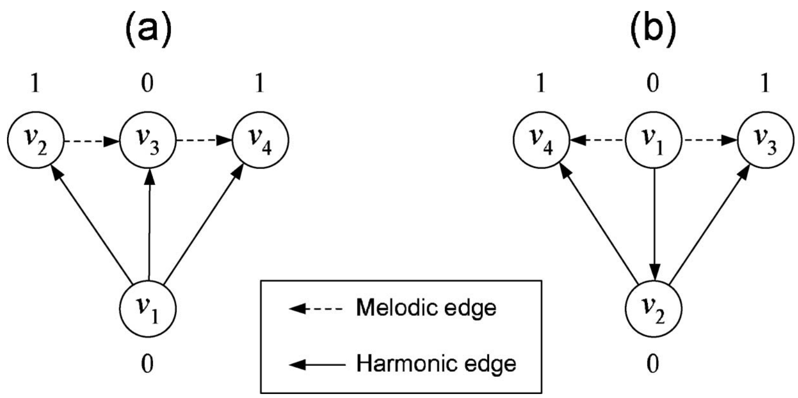


Fig. 5. Two total ordering assignments to the same musical graph. (a) A directed Hamiltonian path exists. (b) A directed Hamiltonian path does not exist.

ties, break them arbitrarily. Under this total ordering assignment, a directed Hamiltonian path exists from the smallest vertex to the largest vertex in any CRP subgraph.

Proof. Given the vertices of a CRP subgraph $V' = \{v_{i_1}, v_{i_2}, \ldots, v_{i_n}\}$ and $v_{i_1} < v_{i_2} < \cdots < v_{i_n}$. As the path starts from the smallest vertex v_{i_1}, we need to prove that any vertex v_{i_k} can be reached from $v_{i_{k-1}}$ for $2 \le k \le n$. The vertex $v_{i_{k-1}}$ is either in the same maximal melodic path of v_{i_k} or in another path. If $v_{i_{k-1}} \not\sim v_{i_k}$, according to the definition of a CRP graph, there exists a harmonic edge between $v_{i_{k-1}}$ and v_{i_k}. If $v_{i_{k-1}} \sim v_{i_k}$, we can use a simple proof by contradiction to show that there exists a melodic edge between $v_{i_{k-1}}$ and v_{i_k}. Suppose otherwise, i.e. suppose that there is no melodic edge between $v_{i_{k-1}}$ and v_{i_k}. Then, there must exist a vertex v_{i_*} in the same maximal melodic path of v_{i_k} such that $v_{i_{k-1}} < v_{i_*} < v_{i_k}$, which contradicts our total ordering assumption. Hence, a directed Hamiltonian path exists.

Taking advantage of the total ordering of vertices, we redefine the harmonic choice set C_t^{h} and the auxiliary set $pathAdj(X_1{:}t-1)$ below. The definitions of the rejection set D_t, the melodic choice set C_t^{m}, and the choice set C_t are restated for convenience.

$$
\begin{aligned}
C_t^{\mathrm{h}} &= \left\{ v \in \left(C_{t-1}^{\mathrm{h}} - \{x_{t-1}\}\right) \mid vx_{t-1} \in E_{\mathrm{h}}^* \right. \\
&\qquad \left. \text{and } v > x_{t-1} \right\} \\
pathAdj(X_{1:t-1}) &= \left\{ v \in \left(V^* - X_{1:t-1}\right) \mid vx \in E_{\mathrm{m}}^* \right. \\
&\qquad \left. \text{for some } x \in X_{1:t-1} \text{ and } v > x_{t-1} \right\} \\
D_t &= \left\{ v \in pathAdj(X_{1:t-1}) \mid vx \notin E_{\mathrm{m}}^* \right. \\
&\qquad \left. \text{for some } x \in (X_{1:t-1} - path(x)) \right\} \\
C_t^{\mathrm{m}} &= \left\{ v \in \left(pathAdj(X_{1:t-1}) - D_{t-1}\right) \mid vx \in E_{\mathrm{m}}^* \right. \\
&\qquad \left. \text{for each } x \in (X_{1:t-1} - path(x)) \right\} \\
C_t &= C_t^{\mathrm{h}} \cup C_t^{\mathrm{m}}
\end{aligned}
$$

Algorithm 1. Finding all matched CRP subgraphs

```
ALLMATCHEDCRP(t)
global X, C_t, D_t (t = 1, 2,..., |V*|)
if t = 1 then
    C_t ← V*
    D_t ← φ
else
    C_t ← gAdj*_h(x_{t-1}) ∩ C_{t-1}
    for each v ∈ (pathAdj(X_{1:t-1}) − D_{1:t-1}) do
        if (X_{1:t-1} − path(v)) ⊆ Adj*_h(v) then
            C_t ← C_t ∪ {v}
        else
            D_t ← D_t ∪ {v}
for each x ∈ C_t do
    x_t ← x
    if X_{1:t} = Q then
        output (X_{1:t})
    ALLMATCHEDCRP(t + 1)
```

Table 3. Precomputed sets in Example 7.

v	$Adj_{\mathrm{h}}(v)$	$gAdj_{\mathrm{h}}(v)$	$path(v)$
v_2	$\{v_3, v_4\}$	$\{v_3, v_4\}$	$\{v_2, v_5\}$
v_3	$\{v_2\}$	ϕ	$\{v_1, v_3, v_4, v_6, v_7, v_8, v_9\}$
v_4	$\{v_2\}$	ϕ	$\{v_1, v_3, v_4, v_6, v_7, v_8, v_9\}$
v_5	ϕ	ϕ	$\{v_2, v_5\}$
v_8	ϕ	ϕ	$\{v_1, v_3, v_4, v_6, v_7, v_8, v_9\}$
v_9	ϕ	ϕ	$\{v_1, v_3, v_4, v_6, v_7, v_8, v_9\}$

A new harmonic adjacency set $Adj_{\mathrm{h}}^*(v)$ is defined as follows:

$$Adj_{\mathrm{h}}^*(v) = \left\{ w \in V^* \mid vw \in E_{\mathrm{h}}^* \right\}$$

The harmonic choice set can be computed more efficiently if we define the set of all vertices greater than and harmonically adjacent to a vertex v:

$$gAdj_{\mathrm{h}}^*(v) = \{ w \in V^* \mid vw \in E_{\mathrm{h}}^* \text{ and } w > v \}$$

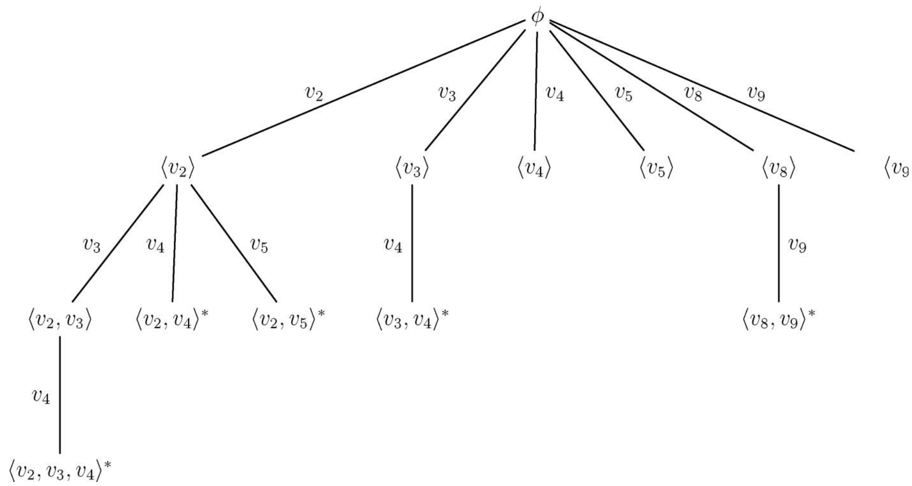


Fig. 6. The backtrack tree in Example 7. Mathced CRP subgraphs are indicated with a "*".

The set $gAdj^*_{\mathrm{h}}(v)$ can be precomputed before running the backtracking algorithm. Then, we have

$$C^{\mathrm{h}}_t = gAdj^*_{\mathrm{h}}(x_{t-1}) \cap C^{\mathrm{h}}_{t-1} = gAdj^*_{\mathrm{h}}(x_{t-1}) \cap C_{t-1}$$

In addition to $gAdj^*_{\mathrm{h}}(v)$, the sets $Adj^*_{\mathrm{h}}(v)$ and $path(v)$ can also be precomputed before running the backtracking algorithm. The backtracking algorithm of finding all matched CRP subgraphs is shown in Algorithm 1. The algorithm starts with ALLMATCHEDCRP(1).

Example 7. *The application of Algorithm 1 on the musical graph in Figure 4 is illustrated below. The precomputed sets are shown in Table 3. The backtrack tree is depicted in Figure 6. Every node except the root in the backtrack tree is a CRP subgraph. Thus, the graph contains 12 CRP subgraphs, in which five of them are matched CRP subgraphs. The matched CRP subgraphs are marked with a "*".*

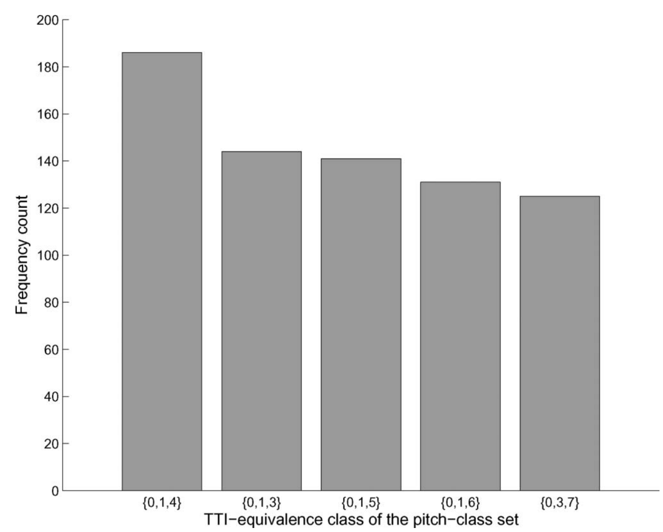


Fig. 7. The top 5 frequency counts of all TTI-equivalence classes.

6. Experiments

6.1 Experiment 1

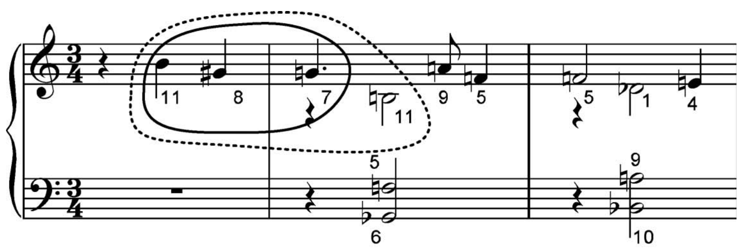
In the first experiment, we compare our algorithm outputs with the musicologists' analysis. Musicologists are usually interested in a group of pitch-class sets instead of one pitch-class set. For example, if the query Q is the pitch-class set {0,1,4}, they are also interested in the shifted version of Q such as $\{0+1, 1+1, 4+1\} = \{1,2,5\}$; and the mirrored versions of Q and its shifted versions such as $\{0,-1,-4\} = \{0,11,8\}$, $\{-1,-2,-5\} = \{11, 10,7\}$, and so on. The group of these pitch-class sets is called *transpositional and inversional equivalence class*, or *TTI-equivalence class* for short. All 4017 pitch-class sets can be grouped into 216 equivalence classes. The details can be found in Straus (2000).

For Arnold Schoenberg's *Klavierstücke*, Op. 11, No. 1, a widely studied piece, it has been demonstrated that the TTI-equivalence class of the pitch-class set {0,1,4} is extensively used (Wittlich 1974; Perle 1991, Williams, 1997; Straus, 2000). In our experiment, we count and search for all 216 TTI-equivalence classes of pitch-class sets. The five TTI-equivalence classes with highest frequency count are shown in Figure 7. The TTI-equivalence class of the pitch-class set {0,1,4} has the highest frequency count. This matches the analysis by the musicologists.

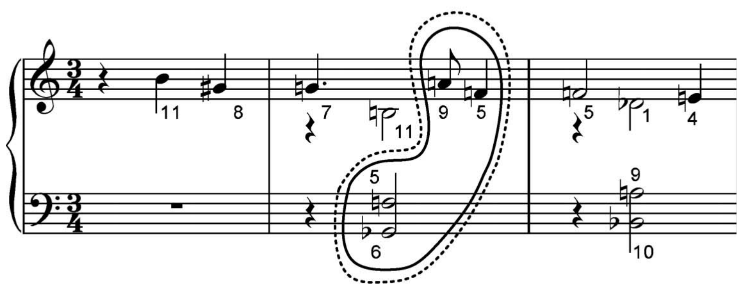
6.2 Experiment 2

In the second experiment, we compare the TTI-equivalence class of the pitch-class set {0,1,4} found by the graph-theoretical approach with those analysed by the musicologist. In Wittlich (1974), Wittlich shows the complete analysis of pitch-class sets in bars 1 to 33 and 53 to 64 of Schoenberg's *Klavierstücke*, Op. 11, No. 1, in which 31 answers of the TTI-equivalence class of the pitch-class set {0,1,4} are located. We investigate whether the graph-theoretical approach can find these 31 answers.

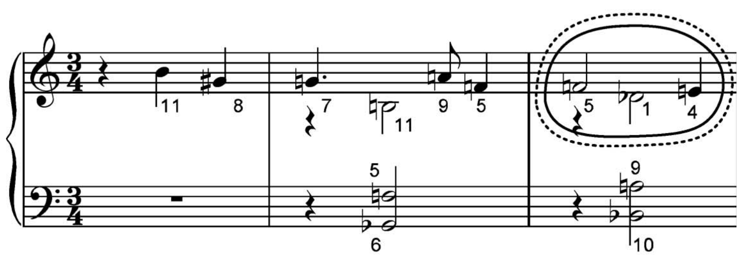
Before showing the experimental results, we explain some definitions and evaluation criteria as follows. The Wittlich's analysis and the outputs of the graph-theoretical approach correspond to two answer sets, which contain answers. An answer A_i in an answer set A is a set of musical notes M_i matching a pitch-class set query Q_i. We denote $A = \{A_i | i = 1,2,\ldots\}$ where $A_i = (Q_i, M_i)$. The Wittlich's analysis consists of such 31 answers, which are written as $A^{\mathrm{m}} = \{(Q_1^{\mathrm{m}}, M_1^{\mathrm{m}}), \ldots (Q_{31}^{\mathrm{m}}, M_{31}^{\mathrm{m}})\}$ where Q_i^{m} is in the TTI-equivalence class of the pitch-class set {0,1,4} and M_i^{m} is the note set matching the pitch-class set Q_i^{m}.



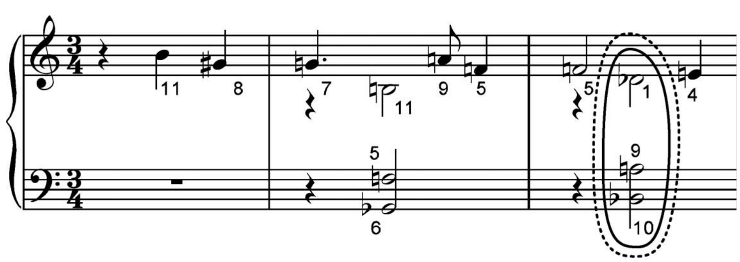
(a) The hit of the pitch-class set $\{7, 8, 11\}$.



(b) The hit of the pitch-class set $\{5, 6, 9\}$.



(c) The hit of the pitch-class set $\{1, 4, 5\}$.



(d) The hit of the pitch-class set $\{1, 9, 10\}$.

Fig. 8. The hits of the TTI-equivalence class of the pitch-class set {0,1,4} in the first three bars of Schoenberg's *Klavierstücke*, Op. 11 No. 1. The set of notes surrounded by a solid line is the answer identified by the graph-theoretical approach. The set of notes surrounded by a dotted line is Wittlich's answer in Wittlich (1994). The hit in (a) is an approximate hit and the hits in (b), (c) and (d) are exact hits.

An output answer is a hit when (i) its note set contains some notes of an answer in Wittlich's analysis, (ii) both the output answer and Wittlich's answer match the same pitch-class set query, and (iii) the intersection between the output answer and Wittlich's answer contains all pitch classes in the pitch-class set query. It is defined below: an output answer $A_i^{\mathrm{o}} = (Q_i^{\mathrm{o}}, M_i^{\mathrm{o}})$ is a hit if there exists a Wittlich answer $A_j^{\mathrm{m}} = (Q_j^{\mathrm{m}}, M_j^{\mathrm{m}})$ such that $Q_i^{\mathrm{o}} = Q_j^{\mathrm{m}}, M_i^{\mathrm{o}} \cap M_j^{\mathrm{m}} \neq \phi$, and the pitch classes in $M_i^{\mathrm{o}} \cap M_j^{\mathrm{m}}$ equals Q_i^{o}. If the note set of Wittlich's answer and that of the output answer are the same, i.e. $M_i^{\mathrm{o}} = M_j^{\mathrm{m}}$, the hit is an exact hit (Figures 8(b), (c) and (d)); otherwise, it is an approximate hit (Figure 8(a)). If an output answer is an approximate hit, its note set is different from Wittlich's one. The number of note differences between an approximate hit and its corresponding Wittlich's answer is defined as follows: given that an output answer $A_i^{\mathrm{o}} = (Q_i^{\mathrm{o}}, M_i^{\mathrm{o}})$ is a hit of Wittlich's answer $A_i^{\mathrm{m}} = (Q_j^{\mathrm{m}}, M_j^{\mathrm{m}})$, the number of note differences is $|M_i^{\mathrm{o}} - M_j^{\mathrm{m}}| + |M_j^{\mathrm{m}} - M_i^{\mathrm{o}}|$. For example, the number of note differences in Figure 8(a) is 1. The output answers in Figures 8(b), (c) and (d) are exact hits so the numbers of note differences are all zero. A miss is a Wittlich's answer for which there is no corresponding hit in the output answer set. A false alarm is an output answer which is not a hit. The definitions of other evaluation criteria and the experimental results are described in Table 4. The musical score of the first three bars and its musical graph generated by the stream segregation algorithm are shown in Figure 9. The hits in the first three bars are depicted in Figure 8.

Table 4. Experimental results of the graph-theoretical approach. The definitions of the evaluation criteria are written next to their names.

Evaluation criteria	Result
No. of answers in Wittlich's analysis (A)	31
No. of answers in the graph-theoretical approach (B)	106
Total hit ($C = D + E$)	28
Exact hit (D)	17
Approximate hit (E)	11
Miss ($A - C$)	3
False alarm ($B - C$)	78
Recall ($F = C/A$)	0.903
Precision ($G = C/B$)	0.264
F-measure $(2 \cdot F \cdot G)/(F + G))$	0.409
No. of note differences (H)	33
Average no. of note differences per hit (H/C)	1.18

The recall value of the graph-theoretical approach is 0.903. Of Wittlich's 31 answers, 28 are successfully found. Hits show close resemblance to Wittlich's answers, as 17 of them are exact hits (60.7%). The number of note differences per hit is 1.18 on average.

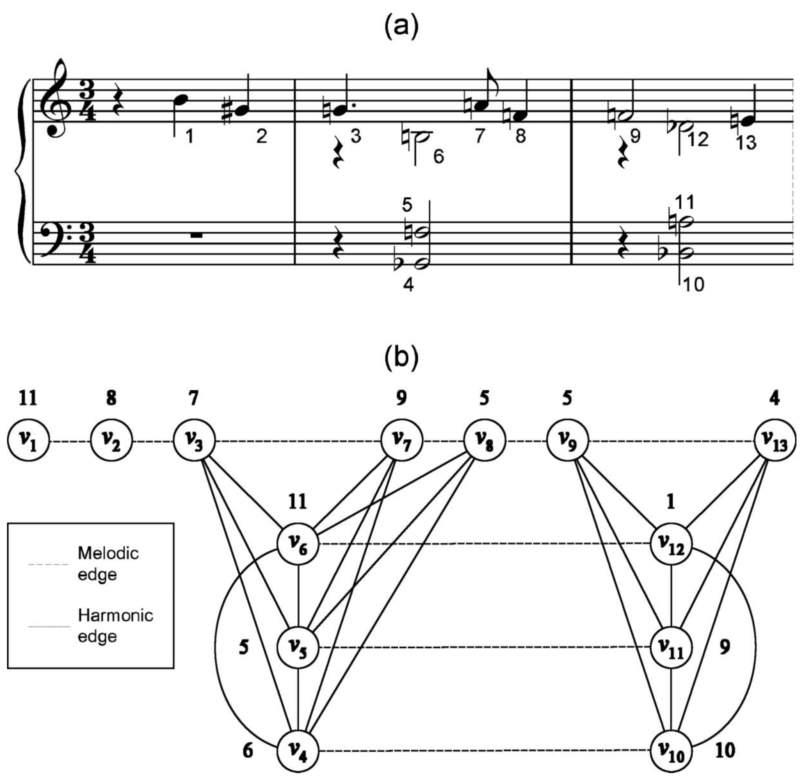
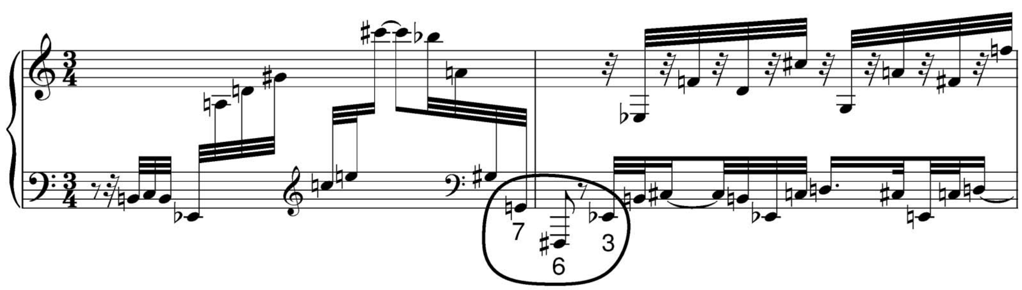


Fig. 9. (a) The musical score of Schoenberg's *Klavierstücke*, Op. 11, No. 1, bars 1–3. The number next to a note is the index of its vertex. (b) The corresponding musical graph. This graph is generated by the stream segregation algorithm introduced in Section 2.

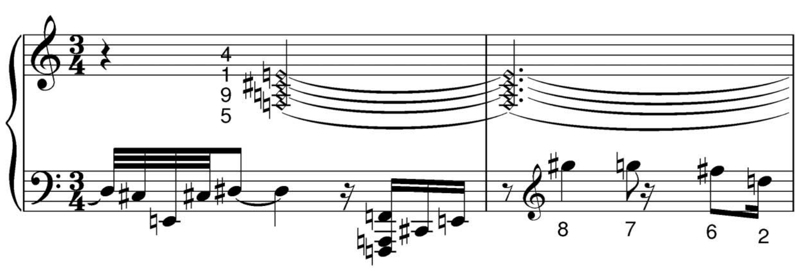
The low F-measure (0.401) can largely be explained by the low precision (0.264). Several reasons may account for the precision value.

1. In Wittlich (1974), he also includes other pitch-class sets other than the TTI-equivalence class of the set {0,1,4} in his analysis. In fact, 13 of the 78 false alarms are pitch classes forming part of larger sets identified by Wittlich. In Figure 10(a), the answer of the set {6,9,10} (one of TTI-equivalents of {0,1,4}) is a subset of Wittlich's answer of the set {2,6,8,9,10,11}. A possible way of limiting false

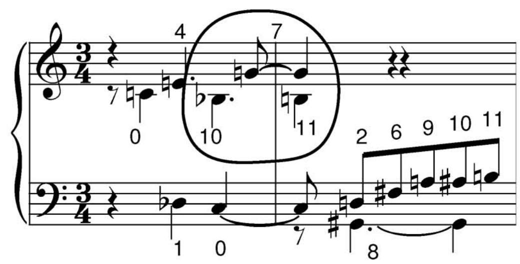
(a) Bars 4–5. The solid line shows the output answer of the pitch-class set $\{6,9,10\}$. The set of the output answer is a subset of a larger set $\{2,6,8,9,10,11\}$ of the Wittlich's answer which is surrounded by the dotted line.



(b) Bars 12-13. The output answer of the pitch-class set $\{3,6,7\}$. There are melodic edges between G♮2 and F♯1, as well as between F♯1 and E♭2.



(c) Bars 14–15. In bar 15, G♯5 in the left hand forms an answer of $\{5,8,9\}$ with F♮3 and A♮3 in the chord, while F♯5 also forms another answer, that of $\{5,6,9\}$, with those two notes in the chord.



(d) Bars 7–8. The output answer of the pitch-class set $\{7,10,11\}$. Although this output answer is not included in Wittlich's analysis, it is regarded as analytically significant in Forte (1981).

Fig. 10. Some false alarms in Schoenberg's *Klavierstücke*, Op. 11 No. 1.

alarms as such is to identify answers which are subsets of some answers of larger pitch-class sets and eliminate them. This is an effective method with the graph-theoretical approach, since the larger sets of all the 13 false alarms are successfully identified. Such an elimination step will remain optional, as musicologists may be interested in the subsets.

2. The performance of the stream segregation algorithm certainly affects the outputs. The stream segregation algorithm introduced in Section 2 tends to create more melodic edges than is required. In Figure 10(b), the notes in the answer of the set {3,6,7} are in a stream. There are melodic edges between G♮2 and F♯1, as well as between F♯1 and E♭2. The melodic edge between F♯1 and E♭2 can be considered an excessive edge. The number of false alarms involving excessive melodic edges is 13. Nevertheless, stream segregation, in itself, is a complicated issue in music perception and is still under research (Gjerdingen, 1994; McCabe & Denham, 1997; Temperley, 2001; Kilian & Hoos, 2002). One possible way of improving algorithm performance is to let the user modify the results of stream segregation before searching. If the user considers the melodic edge between F♯1 and E♭2 in Figure 10(b) excessive and removes it before searching, false alarms as such will not occur.
3. A sustained chord would form answers easily with the shorter notes sounding simultaneously with the chord, as harmonic edges will be formed. In Figure 10(c), the chord with the notes F♮3, A♮3, C♯4 and E♭4 goes on for 9 crotchet beats (from the second beat of bar 14 to the first beat of bar 17). In bar 15 (Figure 10(c)), G♯5 in the left hand forms an answer of {5,8,9} with F♮3 and A♮3 in the chord, while F♯5 also forms another answer, that of {5,6,9}, with those two notes in the chord. The number of false alarms formed as such is 25. Having the effect of sustained chords considered, they may better be isolated – by removing all harmonic edges connected to them, with the duration threshold defined by the user.
4. Discrepancies amongst musicologists are commonplace, even with the same piece of music being analysed according to the same theory. Therefore, musicologists may be interested in the false alarms that remain (27 instances). For example, although the output answer in Figure 10(d) is not included in Wittlich (1974), it is regarded as analytically significant in Forte (1981).

While recall is satisfactory, three of Wittlich's answers are missed. All three are altered repetitions of basically the same pitch classes (Figure 11). The graph-theoretical approach fails to identify the answer in Figure 11 because the three notes in this answer do not form a CRP subgraph. Given any two notes in a CRP subgraph, they

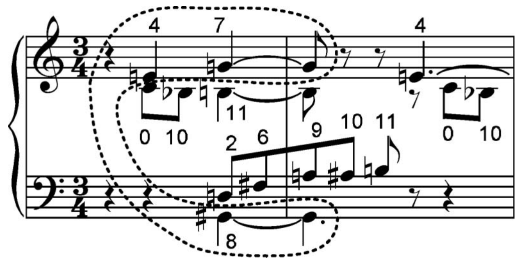


Fig. 11. Bars 4–5. Wittlich's answer of {4,7,8} is missed.

are either (i) connected by harmonic edge or melodic edge, or (ii) in the same melodic path connected by other notes in the CRP subgraph. The E♮4 and the G♮4 are connected by a melodic edge while the G♯2 and the G♮4 are connected by a harmonic edge. As there is no edge between the G♯2 and the E♮4, they belong to different melodic paths and thus cannot form a CRP subgraph. This problem may be solved if the matching conditions become more flexible to include different matching conditions apart from matched CRP subgraphs, although this risks increasing the instances of false alarms.

7. Conclusion

In this paper, the graph-theoretical approach is proposed to facilitate post-tonal music analysis with pitch-class set theory. A musical analysis is modelled as a pattern matching problem, aiming at searching for all matched patterns in the database with a pitch-class set pattern and a music database given. The matching conditions are derived from pitch-class set theory. The graph-theoretical approach also considers how people perceive music by incorporating results of stream segregation into the matching conditions. A piece of music is modelled as a graph, in which each musical note is presented as a vertex and the relationship between a pair of musical notes as an edge. The pairwise relationship is determined by stream segregation. Two kinds of edges are discernible: melodic edge and harmonic edge. Melodic edges capture the relationship between two adjacent events in a stream; while harmonic edges represent the relationship between simultaneous events. Searching for a pitch-class set pattern query is equivalent to searching for a maximal matched CRP (complete *r*-partite path) subgraphs.

We compared the graph-theoretical approach with the musicologists' analysis of Schoenberg's *Klavierstücke*, Op. 11, No. 1. In the first experiment, the dominating TTI-equivalence class of the pitch-class set {0,1,4} was identified and this matches the analysis by the musicologists. In the second experiment, we compared the graph-theoretical approach with an analysis manually done by the musicologist Wittlich (1974) on TTI-equivalence class of {0,1,4}. The graph-theoretical approach achieves high recall (0.903) but low precision (0.264), though 34.6% of the false alarms are potentially musically interesting answers which are not included in Wittlich (1974). Apart from these potentially valuable false alarms, the rest can be reduced by excluding answers which are subsets of larger sets, modifying the results of stream segregation, as well as isolating sustained chords.

Acknowledgements

We are thankful to Eos Cheng, Dennis Wu, and the anonymous reviewers for their valuable comments.

References

Bent, I.D. & Pople, A. (2001). Analysis. In: Sadie, S. (Ed.), *The new Grove dictionary of music and musicians, volume 1* (pp. 526–589). London: Macmillan.

Bregman, A.S. (1990). *Auditory scene analysis: the perceptual organization of sound*. Cambridge, MA: MIT Press.

Brinkman, A.R. (1990). *Pascal programming for music research*. Chicago, IL: The University of Chicago Press.

Castine, P. (1994). *Set theory objects abstractions for computer-aided analysis and composition of serial and atonal music*. Frankfurt am Main: Peter Lang.

Diestel, R. (2000). *Graph theory*, 2nd edn. Berlin: Springer.

Doerksen, J.F. (1999). Segmentation and interpretation in atonal music. In *Proceedings of the International Computer Music Conference (ICMC)*, Beijing, pp. 337–340.

Forte, A. (1966). A program for the analytic reading of scores. *Journal of Music Theory*, *10*, 330–363.

Forte, A. (1981). The magical kaleidoscope: Schoenberg's first atonal masterwork, opus 11, number 1. *Journal of the Arnold Schoenberg Institute*, *5*(2), 127–168.

Gjerdingen, R.O. (1994). Apparent motion in music. *Music Perception*, *11*, 335–370.

Isaacson, E.J. (1998). Simnet: A neural-network model for post-tonal segmentation. Presented at the *Annual Conference of the Society for Music Theory*. Chapel Hill, NC: Society for Music Theory.

Kilian, J. & Hoos, H.H. (2002). Voice separation – a local optimization approach. In *Proceedings of the 3rd International Conference on Music Information Retrieval (ISMIR 2002)*, Paris, pp. 39–46.

Kreher, D.L. & Stinson, D.R. (1999). *Combinatorial algorithms: generation, enumeration, and search*. Boca Raton, FL: CRC Press.

McCabe, S.L. & Denham, M.J. (1997). A model of auditory streaming. *The Journal of the Acoustical Society of America*, *101*(3), 1611–1621.

Perle, G. (1991). *Serial composition and atonality: an introduction to the music of Schoenberg, Berg, and Webern*, 6th edn, revised. Berkeley, CA: University of California Press.

O'Shaughnessy, D. (1987). *Speech communication*. New York: Addison-Wesley.

Straus, J.N. (2000). *Introduction to post-tonal theory*, 2nd edn. Upper Saddle River, NJ: Prentice Hall.

Szeto, W.M. & Wong, M.H. (2006). Stream segregation algorithm for pattern matching in polyphonic music databases. *Multimedia Tools and Applications*, *30*(1), 109–127.

Taylor, S.A. (2001). Setfinder. http://www-camil.music.uiuc.edu/faculty/taylor/setfinder/index.html

Temperley, D. (2001). *The cognition of basic musical structures*. Cambridge, MA: MIT Press.

Tomlin, J. (2003). Java set theory machine. http://www.jaytomlin.com/music/settheory/

Weisstein, E.W. (2005). Eric Weisstein's world of mathematics (mathworld^{TM}). http://mathworld.wolfram.com/

Williams, J.K. (1997). *Theories and analyses of twentieth-century music*. Orlando, FL: Harcourt Brace.

Winsor, P. & DeLisa, G. (1991). *Computer music in C*. Blue Ridge Summit, PA: Windcrest.

Wittlich, G. (1974). Intervallic set structure in Schoenberg's op. 11 no. 1. *Perspectives of New Music*, *13*, 41–55.

Appendix A: Graph theory and its notations

We introduce the notations and the definitions of graph theory summarized from Diestel (2000) and Weisstein (2005). A *graph* is a pair $G=(V, E)$ of a set V of *vertices* and a set E of pairs of vertices called *edges*. For conciseness, we shall not always distinguish strictly between a graph and its vertex set. The number of vertices of $G=(V, E)$ is denoted by $|V|$. The *size* of a graph is the number of its vertices. An edge $\{v, w\}$ is simply expressed as vw or wv. The set of all vertices adjacent to a vertex v is denoted by $Adj(v)$. If all the vertices of G are pairwise adjacent, then G is *complete*. If there is no edge at a vertex, the vertex is a *single isolated vertex*.

We define some operations on graphs as follows. Given two graphs $G=(V, E)$ and $G'=(V', E')$, the union of G and G' is that $G \cup G'=(V \cup V', E \cup E')$, and the intersection of G and G' is that $G \cap G'=(V \cap V', E \cap E')$. If $V' \subseteq V$ and $E' \subseteq E$, G' is a *subgraph* of G, denoted by $G' \subseteq G$. Given a subgraph $G'_1=\left(V'_1, E'_1\right)$ of $G=(V, E)$, the subtraction $G'_2=G-G'_1$ is obtained from G by deleting all the vertices in $V \cap V'_1$ and their incident edges. Given a subgraph G'_1 of G, adding a vertex $v \in \left(V-V'_1\right)$ to G'_1 gives a new subgraph $G'_2=\left(V'_2, E'_2\right)$ where $V'_2=V'_1 \cup \{v\}$ and $E'_2=E'_1 \cup vw$ for each $w \in \left(V'_1 \cap Adj(v)\right)$. This is written as G'_1+v.

A *path* is a graph $P=(V, E)$ of the form $V=\{v_1, v_2, v_3,\ldots,v_n\}$ and $E=\{v_1v_2, v_2v_3,\ldots, v_{n-1}v_n\}$. The *ends* of the path P are the vertices v_1 and v_n. A *subpath* $P'=(V', E')$ of the path P is a path of the form $V'=\{v_i,v_{i+1},\ldots,v_k\} \subseteq V$ and $E'=\{v_i v_{i+1},\ldots,v_{k-1}v_k\} \subseteq E$ where $1 \le i \le k \le n$. A *Hamiltonian path* is a path between two vertices of a graph that visits each vertex exactly once.

A *clique* of G is a complete subgraph of G. A clique H of G is *maximal*[4] if there does not exist a vertex v in $G - H$ such that $H + v$ is a clique. An *r-partite* graph $G = (V, E)$ is a graph, in which V can be partitioned into r partition classes such that every edge has its ends in different partition classes. Vertices in the same partition class must not be adjacent. Each partition class is an independent set. If $r = 2$, the graph is called *bipartite graph*. A *complete r-partite graph* is an r-partite graph such that every two vertices from different partition classes are adjacent. If every partition class has one vertex only, then the complete r-partite graph is a complete graph.

[4] *Cliques* and *maximal cliques* are sometimes confusing because it is not uncommon to refer to cliques as maximal cliques. In this paper, a *clique* means a complete subgraph and a *maximal clique* means a maximal complete subgraph.