Chapter 6

A network flow distance for chords

Although, as we have seen in Chapters 3 and 5, the Earth Mover's Distance (EMD) and the Proportional Transportation Distance (PTD) have quite nice properties, they are not perfect. One particularly noticeable problem with the EMD is the fact that with everything else being equal, if one increases the weight of one point in the lighter point set by a large enough amount, one can reach a point where more points from the other point set must be matched to it. There can be similar effects when one uses the PTD: if one increases the weight proportion that is associated with one point by a large enough amount, more points from the other point set are matched to it. This does not directly correspond to music perception – there is no reason why a more important note should be matched with more notes than a less important note. What is really intended – and also achieved – by attaching more weight to certain notes is to increase their influence on the overall distance. See Figure 6.1 for an example of a flow where some weight flows between two points that are musically unrelated.

It would be nice to have a distance measure that preserves as much of the desirable qualities of transportation distances as possible – in particular, continuity and the triangle inequality at least for some cases –, but avoids the effect of matching an important note with unrelated notes.

Another desirable feature would be to normalize weights as it is done for the Proportional Transportation Distance, but still have the possibility of partial matching in the pitch dimension.

This chapter describes a measure that comes closer to these two goals than either the Earth Mover's Distance or the Proportional Transportation Distance.

6.1 Describing transportation distances with maximum flow/minimum cost network flow problems

The Earth Mover's Distance and Proportional Transportation Distance can both be described as special cases of a network flow distance [59], where a certain total amount of weight has to be moved across a network such that the flow is maximized

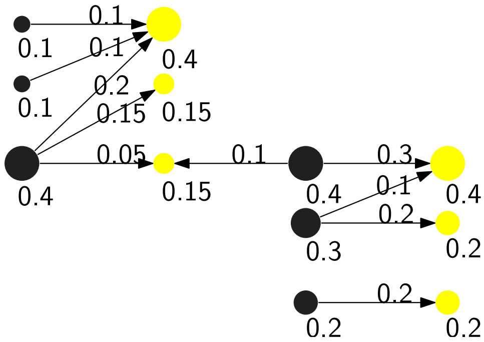

Figure 6.1: An example for unwanted flows with the EMD. Both the dark and the light point set represent two successive chords. To make the flow better visible, the point sets have been pulled apart horizontally; the onset times of the first chords are actually identical. Since the total weight in the two chords differs between the point sets, some of the weight is carried from one chord to the other, which does not make much musical sense. With a local weight normalization, this can be avoided (see Figure 6.5).

and the costs are minimized. The network contains one source node and one sink node. There is a connection from the source node to each node representing a point in the first point set A; these connections each have a capacity that corresponds to the target point's weight, and the cost for using those connections is zero. Each point in point set A is connected to each point in point set B with a connection whose capacity corresponds to the minimum of the two points' weights. Every connection between the point sets A and B has a cost that depends on the ground distance between the two connected points. The points in point set B are each connected to the sink node with connections whose capacities correspond to the weight of the point set in B, and whose costs are zero. The problem of finding a maximum flow at minimum cost through such a network is equivalent to calculating a transportation distance between the two point sets. See Figure 6.2 for an illustration. We simplify the network suggested by Ramon and Bruynooghe [59] by omitting the two extra nodes for carrying surplus weight because this is not needed if we want to normalize both weight sums and enforce that all weight is transported between nodes that represent notes.

It is useful to enforce that all weight flows through the network. If one allows some weight to remain unmatched without a penalty, it is very hard to construct a measure for which the triangle inequality holds. The network flow distance by Ramon et al. [59] obeys the triangle inequality without enforcing all weight to be matched, but it heavily penalizes any unmatched weight by forcing it to flow via an expensive detour in the network. So even without enforcing all weight to be matched, they still force all of it to flow through the network. For the purposes of measuring melodic similarity, penalizing unmatched weight is generally not a good idea, especially for polyphonic music, because the presence or absence of additional voices does not have a very large influence on the identity of a melody. We want to neither allow weight to remain unmatched without penalty (because that would completely break the triangle inequality), nor do we want to penalize unmatched weight like Ramon et al., because that would be inadequate for our application to music.

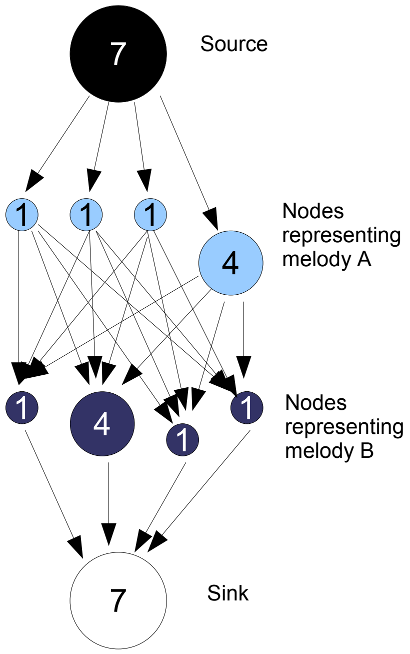

Figure 6.2: Formulation of a transportation distance as a network flow. The black circle at the top is the source, the white one at the bottom is the sink. Source and sink are both connected to each point in one of the two point sets with connections whose capacities correspond to the connected points' weights, and which involve costs of zero. The connections between the two point sets, on the other hand, carry costs that correspond to the ground distance between the connected points.

Enforcing all weight to be matched is also an elegant way of avoiding discontinuities. If, under certain circumstances, we allow some weight to remain unmatched, we would have to take special precautions to ensure that at the point where a point set changes from a situation where all weight is matched to a situation where some weight remains unmatched, the matched amount does not change in a discrete step. It could change in a discrete step, for example, if we would remove a connection with a non-zero capacity from the network. If discontinuities exist, one can always construct point sets that are rather similar but have quite different distances to the same vantage object. Therefore, even if the triangle inequality holds, vantage indexing can become much less useful if there are large discontinuities.

6.2 A network flow distance with normalized weights for chords

6.2.1 First aim: realistic matches for notes

We would like to create a transportation distance that does not have the problem described in the introduction to this chapter, that is, increasing the weight of one particular point should not necessarily lead to it being matched to more points in the other point set, in particular not to points that represent notes with very different onset times.

An obvious idea might be to remove connections according to some rule that specifies under what circumstances notes may be matched with one another. This way, one might, for example, want to modify the network shown in Figure 6.2 such that it is turned into the network shown in Figure 6.3 before calculating the maximum flow with minimum cost.

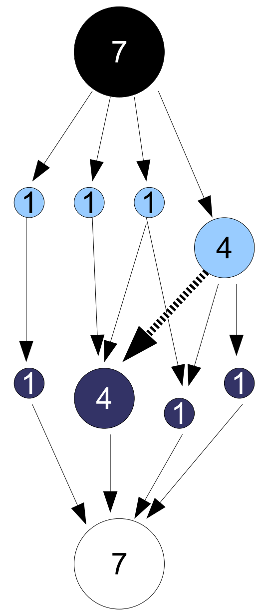

Figure 6.3: A naïve approach to enforcing rules about acceptable matches of notes is to just remove unwanted connections. For example, doing this to the network shown in Figure 6.2 could result in a network like this one. If one would also remove the dashed arrow (because the rightmost long note in the top point set really should not be matched with the last three notes in the bottom point set), solving the flow problem becomes infeasible.

However, there are two problems with this naïve approach of removing unwanted connections: if it is possible to avoid bad matches, the algorithm for finding the maximum flow at minimum cost will always do so anyways (because "bad" connections will always be associated with large ground distances and therefore high costs), and therefore by removing these connections, one does not gain anything.

Secondly, even worse, if the removal of some connections has any effect, it will be the effect that solving the network flow problem is no longer feasible. This would, for example, happen if the algorithm for removing connections would decide that the dashed arrow in Figure 6.3 should be removed. We would still want to move the entire weight of 7 from the source to the sink over the network, but this would not be possible, and therefore there would be no minimum cost, and the distance between the two point sets would be undefined.

Instead of somwhat arbitrarily removing connections, one can also approach the aim of reducing unrealistic flows by ensuring a certain balance in the weight distribution before calculating the optimum flow. The following constraint can do this:

- The proportion of the total weight present between the onset and the offset of any note (or chord) should equal the proportion of the time covered by this note (or chord).

Note that this constraint implies that for monophonic music, the weight simply encodes the duration of a note. For polyphonic music, even with this constraint, one can still use the weight for encoding different degrees of importance for different notes, independent of the note duration.

If this constraint is in place, a single note can still be matched with multiple other notes, but only as long as the durations and the onset times of the involved notes make this plausible.

6.2.2 Second aim: polyphony, fuzzy queries

By adding arcs between nodes that represent the same point set, we can build some more desirable properties into the network flow distance measure. In order to support polyphonic matching of sequences of chords or fuzzy monophonic queries that contain several possibilities for some notes, one can connect all notes that belong to the same chord with arcs of cost zero and a capacity that allows the complete weight of the lighter of the two points to flow to the other point. Notes can be defined to belong to the same chord if their overlap in time exceeds a certain threshold such as 75 % of the duration of the shorter note. Figure 6.4 shows an example.

Overall, we construct networks like this:

1. Represent the two segments of music that are to be compared as weighted point sets, as described in Section 3.1.

2. Identify chords in both point sets and connect any two notes that belong to the same chord with an arc whose cost is zero and whose capacity equals the weight of the lighter of the two points. Notes are said to belong to the same chord if their overlap in time exceeds a certain percentage of the duration of the shorter of the two notes.

3. For each chord, calculate onset and offset time (find the earliest onset time of any note in the chord, and the latest offset time).

4. For each chord, set the total weight to the duration of the chord (the difference of onset time and offset time), preserving the weight proportions within the chord.

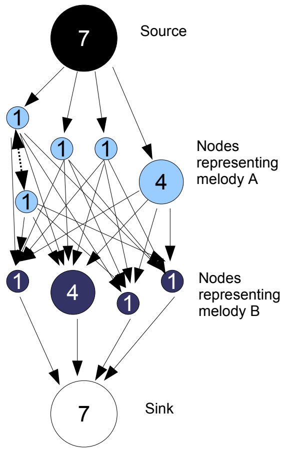

Figure 6.4: An example for how chords can be modeled by adding arcs within point sets. Point set A here contains two notes which overlap in time. In such cases, we add a connection whose capacity allows the entire weight of the lighter of the two points to flow to the other point, and whose cost is zero.

5. For each point set, set the total weight to 1, preserving the weight proportions within the point set.
6. Add a special source node with weight 1 and connect it to every point in the first point set. Each of these arcs has cost zero and a capacity that equals the weight of the node representing a note.
7. Add a special sink node with weight 1 and connect it to every point in the second point in a similar fashion.
8. Connect each point in the first point set with each point in the second point set with a connection whose cost corresponds to the ground distance between the two points and whose capacity equals the weight of the lighter of the two connected points.

6.3 Properties

The new network flow distance with balanced weights and partial matching in the pitch dimension (BWPP) is constructed in a way such that it has the following properties:

1. **The BWPP is continuous under weight modification, the addition or removal of points, and changes in the pitch coordinate.** Neither the weight normalization nor the added arcs within point sets introduce discontinuities if weights are modified or points are added or removed. If weights are modified, this either is undone by the weight normalization, or it has an impact on the overall distance that is proportional to the modification of the weight. If points are added or removed, the change in the overall distance

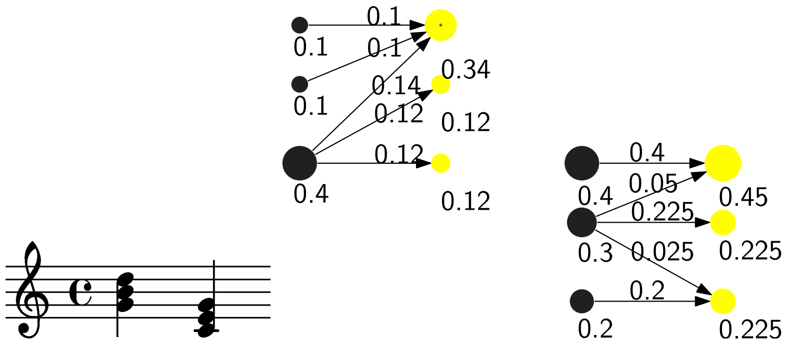

Figure 6.5: By normalizing the weights of chords, we can avoid unwanted flows like in Figure 6.1 without giving up the principle that all weight should be matched. Since flows within a chord do not incur costs, the proposed distance measure would result in a distance of zero between these two pairs of chords, which makes sense because they consist of identical notes.

can be zero, or it is proportional to the weight of the added or removed point. If a point is moved up or down in pitch, this causes changes in the ground distance that are proportional to the change in the coordinate.

2. **The BWPP is not always continuous if the onset time of notes is changed.** If the time coordinate of a point is changed such that it starts or stops being part of a chord, even a small change of the time coordinate can cause a discontinuity because suddenly, all points that are part of the chord in question have one additional possible connection to the other point set, or lose one. However, this kind of discontinuity can only occur if there are chords; as long as all point sets are monophonic, changes in the coordinates of points do not cause discontinuities.

3. **The BWPP obeys the triangle inequality if there are no chords.** Since we normalize the total weight for both point sets, the BWPP obeys the triangle inequality for the same reasons the Proportional Transportation Distance obeys the triangle inequality. However, as soon as we introduce arcs within point sets, it becomes possible to construct point sets for which the BWPP does not obey the triangle inequality anymore. Figure 6.6 shows an example where this occurs.

4. **The BWPP is suitable for fuzzy queries or for matching arbitrary combinations of polyphonic and monophonic music.** Thanks to the added arcs within chords, one can try several alternatives for one note in the query at once. For example, if it is not known what the second note in a monophonic query should be, but there are several possible candidates, one could put all possible notes into a chord where the second note would be, and the distance measure would find matches for all possibilities.

6.3.1 Experimental evaluation with MIREX 2006 data

We created a vantage index using the new distance measure for the "Karaoke" collection of 1000 MIDI files from MIREX 2006 and calculated Average Dynamic

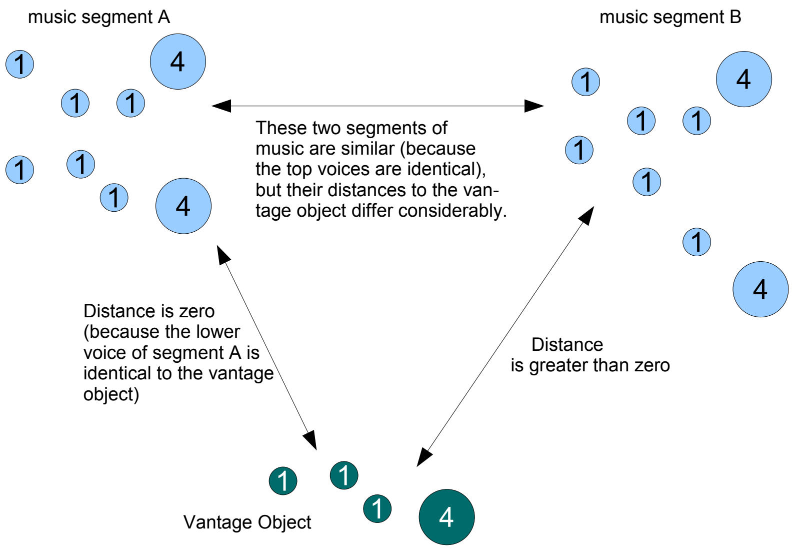

Figure 6.6: An example for how the triangle inequality can be violated for point sets representing polyphonic music.

Table 6.1: Result quality for the MIREX 2006 symbolic melodic similarity "Karaoke" task. ADR at MIREX=Average Dynamic Recall for the measure we submitted to MIREX 2006, ADR with new measure=the ADR score of the measure described in this chapter.

Query number	ADR at MIREX	ADR with new measure
1	1	0.67
2	0.54	0
3	0.56	0
4	1	0.5
5	1	0
Average	0.82	0.23

Recall (ADR) at rank 10 for the same queries that were used at MIREX 2006. The results can be seen in Table 6.1.

Since we only counted true positives that were ranked among the top ten items, we did not count two matches for one query which the new measure found but our MIREX 2006 distance measure did not. The new measure placed them between the tenth and the twentieth position, so they were still ranked higher than 98 percent of the 1000 pieces in the collection. However, this does not change the fact that the new network flow distance performs much worse than the PTD.

An important difference between the new measure and the MIREX version is that at MIREX, all points were created with equal weights, that is, all notes were given exactly the same importance. For this experiment with the new distance measure, however, we gave the points weights that were proportional to the note durations. Therefore, one cannot deduct from this table that the new network distance always performs worse than the PTD. This result does indicate, however, that it was probably a good decision to give all points equal weights at MIREX 2006.

There is still a large class of cases where even with intra-point set weight distributions that are normalized according to the new measure, unwanted flows occur that do not make much musical sense. The next section describes this problem and how it can be solved.

6.4 Using inter-onset time instead of note durations for distributing weights

Even if for every chord (or, for monophonic melodies, for every note), the weight is proportional to the time that is covered by the chord (or note), situations can arise where very unrealistic flows can occur. Figure 6.7 shows an example (two slightly different renditions of a motive from Johannes Brahms, "Ein deutsches Requiem", first movement). The cause of the problem is that imbalances are not prevented by this constraint if the inter-onset intervals of notes are noticeably larger than their durations. With real-world data such as the MIREX Karaoke collection, this happens quite frequently.

The top point set in Figure 6.7 has a much heavier first note than the bottom point set, but all but the last notes have the same weights. Because of this, the weight surplus is forced to flow all the way to the last note of the bottom point set. If weight is forced to flow from the first to the last note like this, the resulting distance depends on the difference in onset times of two largely unrelated notes, which does not make much musical sense. The coordinates and weights for the top point set are (in the format <onset time><pitch><weight>): 0 7.2 0.5/0.75 7.4 0.125/1 7.6 0.25/1.5 7.4 0.25/2 7.2 0.25/2.5 7.1 0.25/3 6.9 0.25/3.75 7.1 0.125. The bottom point set is 0 7.2 0.375/0.8 7.4 0.125/1.25 7.6 0.25/1.75 7.4 0.25/2.25 7.2 0.25/2.5 7.1 0.25/3 6.9 0.25/3.5 7.1 0.25. As we can see, the inter-onset interval is always larger than the note duration.

Weight imbalances can be prevented in a larger number of cases if we replace the constraint from Section 6.2.1 with:

- The proportion of the total weight present between the onset and the offset of any note (or chord) should equal the proportion of the time covered by its inter-onset interval (or, for the last note or chord, its duration).

If we normalize weights for the two point sets from Figure 6.7 by applying this constraint, we get the two new point sets 0 7.2 0.375/0.75 7.4 0.125/1 7.6 0.25/1.5 7.4 0.25/2 7.2 0.25/2.5 7.1 0.25/3 6.9 0.375/3.75 7.1 0.125 (top) and 0 7.2 0.4/0.8 7.4 0.225/1.25 7.6 0.25/1.75 7.4 0.25/2.25 7.2 0.125/2.5 7.1 0.25/3 6.9 0.25/3.5 7.1 0.25 (bottom). We leave everything else equal, but get the much more realistic flow that is shown in Figure 6.8. This flow does not only look more realistic, but it involves much lower costs: 0.167 instead of 0.33 (with the Euclidean distance as ground distance), which is desired because the melodic similarity is very pronounced.

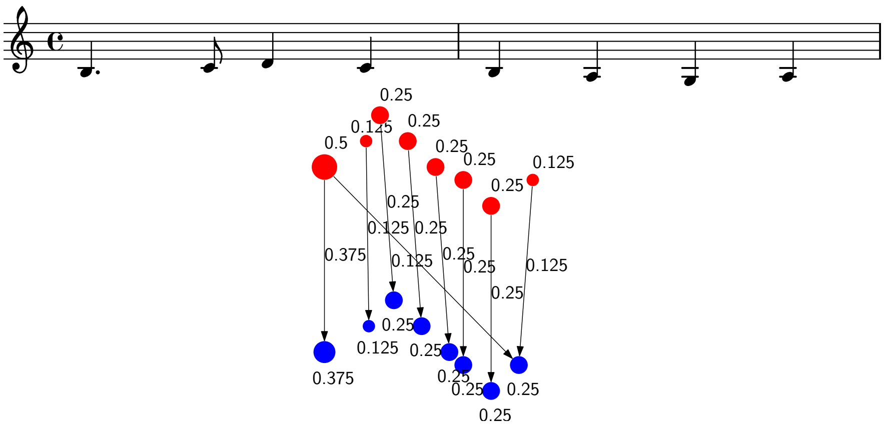

Figure 6.7: Making the weight proportional to the time proportion that each note covers does not always prevent the optimum flow from being unrealistic. The top and bottom point sets are slightly different renditions of the same motive from Brahms' "Ein deutsches Requiem", first movement, and should therefore be recognized as rather similar.

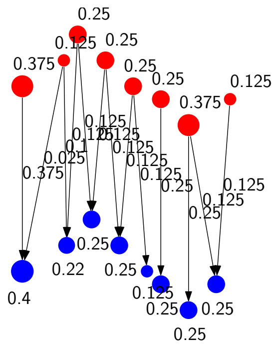

Figure 6.8: By using inter onset intervals instead of note durations for enforcing an even distribution of weight, the problem shown in Figure 6.7 can be avoided.

6.5 Conclusions

By using the more general formulation of network flows, one can construct transportation distances with certain desired properties. One can use weight normalizations, arc capacities and costs, and additional arcs to modify the properties of the resulting dissimilarity measures. We have shown one possible measure that reduces unrealistic matching of notes and that supports fuzzy queries and polyphonic matching. The introduction of chords within which free weight flows are possible, however, means that the triangle inequality does no longer hold.

For making the violations of the triangle inequality less problematic for vantage indexing, one might want to experiment with multiple small groups of vantage objects; for each group, one would search the intersection of rings around vantage objects, as shown in Figure 4.2. This would limit the number of false positives, which, as we have seen in Section 4.4, already drops significantly with small numbers of vantage objects. The false negatives that are introduced by the problem illustrated in Figure 6.6 could then be counteracted by looking at the union of the candidate sets produced by different sets of vantage objects. If the vantage objects in different groups are sufficiently different, the violations of the triangle inequality for any item should differ for each set of vantage objects. A false negative would then only occur if one is unlucky enough to encounter a severe enough violation of the triangle inequality with every group of vantage objects.