Algorithm 1: A generalized McAulay-Quatieri peak-matching algorithm.

Input: the cost matrix $\boldsymbol{C}$
Output: L tuples of indices Γ, or fewer if Δ_{MQ} exceeded
$\Gamma \leftarrow \varnothing$;
for $l \leftarrow 0$ *to* $L-1$ **do**
  $\Gamma_l = \underset{[0,\ldots,M_0-1]\times\ldots\times[0,\ldots,M_{K-1}-1]\setminus\Gamma}{\arg\min} \boldsymbol{C}$;
  if $\exists i,j \in \Gamma_l : \mathcal{D}(\theta_i,\theta_j) > \Delta_{MQ}$ **then**
    return Γ
  end
  $\Gamma \leftarrow \Gamma \cup C_{\Gamma_l}$;
end
return Γ

smaller costs to be chosen in successive iterations. This algorithm does not allow for that. In other terms, the algorithm does not find a set of pairs that represent a globally minimal sum of costs. Furthermore, the algorithm does not scale well: assuming equal numbers of parameter sets in all frames, the search space grows exponentially with K. Nevertheless, the method is simple to implement, computationally negligible when K is small, and works well with a variety of audio signals such as speech [1] and music [9].

3. L BEST PATHS THROUGH A LATTICE VIA LINEAR PROGRAMMING (LP)

In this section we show how to find L paths through a lattice of K frames such that the sets of nodes on each path are disjoint. The kth frame of the lattice contains N_k nodes for a total of $M = \sum_{k=0}^{K-1} N_k$ nodes.

Similar to the McAulay-Quatieri method we define the cost Δ_{LP} as the limiting cost under which the connection between two nodes will be considered in the LP method.

The solution vector $\boldsymbol{x}$ to the linear program shall indicate the presence of a connection between a pair of nodes by having an entry equal to 1 and otherwise have entries equal to 0. To enumerate the set of possible connection-pairs we define

$$\rho = \{(i,j) : \mathcal{D}(\theta_i,\theta_j) \leq \Delta_{\mathrm{LP}}, 0 \leq i < M, 0 \leq j < M, i \neq j\} \tag{1}$$

The cost vector of the objective function is then

$$\boldsymbol{c}_\rho = \{D(\theta_i,\theta_j)\forall(i,j) \in \rho\} \tag{2}$$

and the length of $\boldsymbol{c}_\rho$ is $\#\rho = \#\boldsymbol{c}_\rho = P$, in other words, P pairs of nodes. For convenience we define a bijective mapping $\mathcal{B} : \rho \rightarrow [0,\ldots,M-1]$ giving the index in $\boldsymbol{x}$ of the pair $p \in \rho$. For the implementation considered in this paper, $\mathcal{D}(\theta_i,\theta_j) = \infty$ for all i,j not in adjacent frames and so P will be no larger than $(K-1)N^2$ (assuming the same number of nodes N in each frame).

The total cost of the paths in the solution is then calculated through the inner product $\boldsymbol{c}_\rho^T\boldsymbol{x}$. To obtain $\boldsymbol{x}^*$ that represents L disjoint paths we must place constraints on the structure of the solution. Some of the constraints presented in the following are redundant but the redundancies are kept for clarity; later we will show which constraints can be removed without changing the optimal solution $\boldsymbol{x}^*$.

All nodes in $\boldsymbol{x}^*$ will have at most one incoming connection or otherwise no connections, a constraint that can be enforced through the following linear inequality: define $\boldsymbol{A}^{\mathrm{I}} \in \mathbb{R}^{R_{\mathrm{I}}\times P}$ with $R_{\mathrm{I}} = \sum_{k=1}^{K-1} N_k$, the number of nodes in all the frames excluding the first. We sum all the connections into the node $r_{\mathrm{I}} + N_0$ represented by the respective entry in $\boldsymbol{x}$ through an inner product with the r_{I}th row in $\boldsymbol{A}^{\mathrm{I}}$ and require that this sum be between 0 and 1, i.e.,

$$\boldsymbol{A}^{\mathrm{I}}_{r_{\mathrm{I}},\mathcal{B}(p)} = \begin{cases} 1 & \text{if } p_1 = r_{\mathrm{I}} + N_0 \\ 0 & \text{otherwise} \end{cases}, 0 \leq r_{\mathrm{I}} < R_{\mathrm{I}}, p \in \rho \tag{3}$$

and

$$\boldsymbol{0} \leq \boldsymbol{A}^{\mathrm{I}}\boldsymbol{x} \leq \boldsymbol{1} \tag{4}$$

Similarly, to constrain the number of outgoing connections into each node, we define $R_{\mathrm{O}} = \sum_{k=0}^{K-2} N_k$ and $\boldsymbol{A}^{\mathrm{O}} \in \mathbb{R}^{R_{\mathrm{O}}\times P}$ with

$$\boldsymbol{A}^{\mathrm{O}}_{r_{\mathrm{O}},\mathcal{B}(p)} = \begin{cases} 1 & \text{if } p_0 = r_{\mathrm{O}} \\ 0 & \text{otherwise} \end{cases}, 0 \leq r_{\mathrm{O}} < R_{\mathrm{O}}, p \in \rho \tag{5}$$

and

$$\boldsymbol{0} \leq \boldsymbol{A}^{\mathrm{O}}\boldsymbol{x} \leq \boldsymbol{1} \tag{6}$$

To forbid breaks in the paths it is required that the number of incoming connections into a given node equal the number of outgoing connections for the $R_{\mathrm{B}} = \sum_{k=1}^{K-2} N_k$ nodes potentially having both incoming and outgoing connections.

$$\boldsymbol{A}^{\mathrm{B}}_{r_{\mathrm{B}}} = \boldsymbol{A}^{\mathrm{B}}_{r_{\mathrm{B}}} - \boldsymbol{A}^{\mathrm{B}}_{r_{\mathrm{B}}+N_0} \text{ for rows } 0 \leq r_{\mathrm{B}} < R_{\mathrm{B}} \tag{7}$$

and

$$\boldsymbol{A}^{\mathrm{B}}\boldsymbol{x} = \boldsymbol{0} \tag{8}$$

Finally we ensure that there are L paths by counting the number of connections in each frame and constraining this sum to be L. We choose arbitrarily to count the number of outgoing connections by summing rows of $\boldsymbol{A}^{\mathrm{O}}$ into rows of $\boldsymbol{A}^{\mathrm{C}} \in \mathbb{R}^{(K-1)\times P}$

$$\boldsymbol{A}^{\mathrm{C}}_{r_{\mathrm{C}}} = \sum_{k=a}^{b} \boldsymbol{A}^{\mathrm{O}}_{k} \tag{9}$$

with $a = \sum_{j=0}^{r_{\mathrm{C}}} N_j$ and $b = \sum_{j=0}^{r_{\mathrm{C}}+1} N_j$ and

$$\boldsymbol{A}^{\mathrm{C}}\boldsymbol{x} = L\boldsymbol{1} \tag{10}$$

As stated above, some of these constraints are redundant and can be removed. Indeed, we have $\boldsymbol{0} \leq \boldsymbol{x} \leq \boldsymbol{1}$, therefore we will always have $\boldsymbol{A}^{\mathrm{I}}\boldsymbol{x} \geq 0$ and $\boldsymbol{A}^{\mathrm{O}}\boldsymbol{x} \geq 0$. Furthermore, all but the last row of (10) can be seen as constructed from linear combinations of rows of (8) and the last row of (10) so we only require $\boldsymbol{A}^{\mathrm{C}}_{K-2}\boldsymbol{x} = L$. Finally we always have $\boldsymbol{x} \leq \boldsymbol{1}$ because of the constraint that there be a maximum of 1 incoming and outgoing connection from each node.

The complete LP to find the L best disjoint paths through a lattice described by node connections ρ is then

$$\min_{\boldsymbol{x}} \boldsymbol{c}_\rho^T\boldsymbol{x}$$

subject to

$$\boldsymbol{G}\boldsymbol{x} = \begin{bmatrix} \boldsymbol{A}^{\mathrm{I}} \\ \boldsymbol{A}^{\mathrm{O}} \\ -\boldsymbol{I} \end{bmatrix} \boldsymbol{x} \leq \begin{bmatrix} \boldsymbol{1} \\ \boldsymbol{1} \\ \boldsymbol{0} \end{bmatrix}$$
$$\boldsymbol{A}\boldsymbol{x} = \begin{bmatrix} \boldsymbol{A}^{\mathrm{B}} \\ \boldsymbol{A}^{\mathrm{C}}_{K-2} \end{bmatrix} \boldsymbol{x} = \begin{bmatrix} \boldsymbol{0} \\ L \end{bmatrix} \tag{11}$$

where $\boldsymbol{I}$ is the identity matrix. A proof that the solution $\boldsymbol{x}^*$ will have entries equal to either 0 or 1 can be found in [10, p. 167].