Dynamic Bayesian Networks for multi-band automatic speech recognition

Khalid Daoudi, Dominique Fohr, Christophe Antoine

Article Submitted to Computer Speech and Language

Dynamic Bayesian networks for multi-band automatic speech recognition

Khalid Daoudi, Dominique Fohr and Christophe Antoine

INRIA-LORIA
Speech Group (www.loria.fr/equipes/parole)
B.P. 101 - 54602 Villers les Nancy. France.
email: daoudi,fohr,antoine@loria.fr

Abstract

This paper presents a new approach to multi-band automatic speech recognition which has the advantage to overcome many limitations of classical muti-band systems. The principle of this new approach is to build a speech model in the time-frequency domain using the formalism of dynamic Bayesian networks. In contrast to classical multi-band modeling, this formalism leads to a probabilistic speech model which allows communications between the different sub-bands and, consequently, no recombination step is required in recognition. We develop efficient learning and decoding algorithms both for isolated and continuous speech recognition. We present illustrative experiments on isolated and connected digit recognition tasks. These experiments show that the this new approach is very promising in the field of noisy speech recognition.

1. Introduction

State-of-the-art automatic speech recognition (ASR) systems are based on probabilistic modeling of the speech signal using Hidden Markov Models (HMMs). These models lead to the best recognition performances in ideal "lab" conditions or for easy tasks. However, in real word conditions of speech processing (noisy environment, spontaneous speech, non-native speakers...), the performance of HMM-based ASR systems can decrease drastically and their use becomes limited. One of the major reasons for this discrepancy is the fact that classical HMM's parameterization and modeling fail to capture some acoustic phenomena which are specific to speech. For instance, while speech temporal dynamics are well captured by HMMs, the frequency dynamics (which are phonetically very informative) are weakly modeled in classical HMM-based systems.

Recently, a new approach to ASR which attempts to add a frequency "dimension" in speech modeling, known as *multi-band* speech recognition, has been proposed [5, 14, 21]. This approach takes its origin in an extensive study done by Harvey Fletcher [15] on how humans process and recognize speech. Basically speaking, this

study (reviewed by Jont B. Allen in [1]) suggests that the human auditory system processes speech *locally* in the time-frequency domain before recognition. The general approach to multi-band speech recognition is to divide the time-frequency domain into several sub-bands, then each sub-band is independently modeled by a HMM. The recognition scores in the sub-bands are then fused with some recombination module. This approach has also been motivated by the desire to improve robustness to additive noise, particularly band-limited noise. Indeed, in classical systems the full frequency band is *globally* processed in order to extract speech features, thus the resulting acoustic vectors are all corrupted even if the noise covers only a small frequency sub-band. Using the multi-band local frequency processing, only the information extracted from the noisy sub-band will be corrupted, the remaining non-corrupted information can be then exploited for recognition.

Definitely the multi-band principle (i.e. local processing in the time-frequency domain) is very attractive because it attempts to mimic the behavior of the human auditory system and it can lead to noise-robust systems. However, the classical approach (described above) to exploit this principle is far from being optimal. For instance, the sub-bands are assumed mutually independent which is an unrealistic hypothesis. Moreover, the information contained in one sub-band is not discriminative in general. In addition, the recombination step can be a very difficult task, particularly in continuous speech recognition.

The scope of this paper is to propose a new approach to multi-band speech recognition which has the advantage to overcome *all* the limitations (mentioned above) of the classical multi-band (CMB) approach. In the latter, the fundamental weakness is the fact that sub-bands modeling is *independent*, the basic idea behind our new approach is to render *dependent* such modeling. A way to do so is to create "communications" or "interactions" between the different HMMs that model the different sub-bands. For this propose, we use the formalism of *Bayesian networks* (BNs) which is an appropriate framework for our goal for two reasons. First, through meaningful graphical representations, Bayesian networks has the advantage to provide a natural tool to represent interactions and dependencies between variables of a given system. Second, by exploiting conditional independence between system variables, they introduce some "modularity" in large-scale problems in order to split them up into small and tractable problems. Consequently, Bayesian networks not only provide an attractive tool for modeling complex systems, but also lead to efficient general-purpose algorithms.

After Judea Pearl's pioneering work [29], Bayesian networks have emerged as a powerful formalism unifying many concepts of probabilistic modeling widely used in statistics, artificial intelligence, signal processing and other fields. For example, HMMs, mixture models, and Kalman filters are all particular instances of the more general BNs formalism. BNs have then become a very popular framework for reasoning under uncertainty and have been widely used in expert systems design and decision making systems. However, the use of BNs in automatic speech recognition has gained attention only very recently [2, 3, 4, 9, 10, 12, 38, 40, 37, 39]. This paper presents a new multi-band system which relies on a "uniform" time-frequency speech model. Namely, instead of considering an independent HMM for each sub-band (as in the CMB approach), we build a more complex but unique dynamic Bayesian net-

work on the time-frequency domain by "coupling" all the HMMs associated with the different sub-bands. We develop the learning and decoding algorithms corresponding to this new speech model and carry out illustrative experiments to show the potential of this new multi-band approach. The paper is organized as follows. In the next section, we give a brief introduction to Bayesian networks. In order to make the paper self-contained, we present in section 3 the inference algorithms that we use in learning and decoding. In section 4, we present the classical approach to multi-band ASR. In section 5, we present the principle and the algorithmic details of our new approach to multi-band ASR. In section 6 and 7, we show how to apply our approach in isolated and continuous speech recognition and present *illustrative* experiments on an isolated and connected digit recognition task.

2. Bayesian networks

During the last decade, Bayesian networks (and probabilistic graphical models in general) have become very popular in artificial intelligence (and other fields) due to many breakthroughs in several aspects of inference and learning. The literature is now extremely rich in papers and books dealing with the theory and applications of BNs, among which we refer to [8, 6] for a very good introduction. The formalism of probabilistic graphical models (PGMs) is well summarized in the following quotation by M. Jordan [23]:

"Graphical models are a marriage between probability theory and graph theory. They provide a natural tool for dealing with two problems that occur throughout applied mathematics and engineering - uncertainty and complexity - and in particular they are playing an increasingly important role in the design of machine learning algorithms. Fundamental to the idea of a graphical model is the notion of modularity - a complex system is built by combining simpler parts. Probability theory provides the glue whereby the parts are combined, ensuring that the system as whole is consistent, and providing ways to interface models to data. The graph theoretic side of graphical models provides both an intuitively appealing interface by which humans can model highly-interacting sets of variables as well as a data structure that lends itself naturally to the design of efficient general-purpose algorithms."

More precisely, given a system of random variables (r.v.), a PGM consists in associating a graphical structure to the joint probability distribution of this system. The nodes of this graph represent the r.v., while the edges encode the (in)dependencies which exist between these variables. One distinguishes three types of graphs: directed, undirected and those for which the edges are a mixture of both. In first case, one talks about *Bayesian networks*, in the second case, one talks about *Markov random fields*, and in the third case one talks about *chain networks*. PGMs have two major advantages:

- They provide a natural and intuitive tool to illustrate the dependencies which exist between variables. In particular, the graphical structure of a PGM clarifies the conditional independencies embedded in the associated joint probability distribution.
- By exploiting these conditional independencies, they provide a powerful setting

to specify efficient inference algorithms. Moreover, these algorithms may be specified automatically once the initial structure of the graph is determined.

So far, the conditional independencies semantics (or Markov properties) embedded in a PGM are well-understood for Bayesian networks and Markov random fields. For chain networks, these are still not well-understood. In our current research, given the causal and dynamic aspects of speech, Bayesian networks (BNs) are of particular interest to us. Indeed, thanks to their structure and Markov properties, BNs are well-adapted to interpret causality between variables and to model temporal data and dynamic systems. In addition, not only HMMs are a particular instance of (dynamic) BNs, but also the Viterbi and Forward-Backward algorithms (which made the success of HMMs in speech) are particular instances of generic inference algorithms associated to BNs [31]. This shows that BNs provide a more general and flexible framework than the HMMs paradigm which has ruled ASR for the last three decades.

Formally, a (static) Bayesian network has two components: a directed acyclic graph S and a numerical parameterization Θ. Given a set of random variables $X = \{X_1, ..., X_N\}$ and $P(X)$ its joint probability distribution (JPD), the graph S encodes the conditional independencies which (are supposed to) exist in the JPD. The parameterization Θ is given in term of conditional probabilities of variables given their parents. Once S and Θ are specified, the JPD can be expressed in a factored way as*

$$P(x) = \prod_{i=1}^{N} P(x_i | pa(x_i)) \tag{1}$$

where $pa(x_i)$ denotes an outcome of the parents of X_i. The conditional independence semantics (or Markov properties) of a BN imply that, conditioned on its parents, a variable is independent of all the other variables except its descendants.

Dynamic Bayesian networks (DBNs) extend the BN representation to dynamic processes. This representation encodes the beliefs about possible trajectories of the process. Consider a time evolving set $X[t] = \{X_1[t], \ldots, X_N[t]\}$ of variables. A DBN encodes the joint probability distribution of these variables in a finite time interval $[0, T]$. In general, this JPD can be encoded in a huge static BN with $T \times N$ variables with (possibly) different structure and parameters for each time slice. If the underlying process is stationary, then the independence assertions and the associated conditional probabilities are identical for each time slice t. In this case, the repeating structure and parameters can be encoded with a static BN in a single time slice. From this point of view, it is obvious that a HMM is a particular DBN as shown in Figure 1. Contrarily to the usual state transition diagram, in the DBN representation each node H_t (resp. O_t) is a random variable whose outcome indicates the state occupied (resp. the observation vector) at time t. Time is thus made explicit and arrows linking the H_t must be understood as "causal influences" (not as state transitions). It is this representation of HMMs that we shall use in the rest of the paper.

In the next section we present the algorithms we use to infer our acoustic models.

*In the whole paper, upper-case (resp. lower-case) letters are used for random variables (resp. outcomes).

This section is introduced to make the paper self-contained. Thus, it can be skipped by readers who are not interested in the algorithmic details. On the other hand, those who are interested in these aspects may find the description too short. We advise them the very nice tutorials on Bayesian networks [16, 31] and also the very interesting thesis [28].

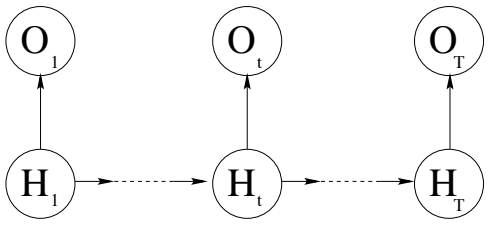

Figure 1: a HMM represented as a dynamic Bayesian network

3. Inference algorithms for Bayesian networks

Once a Bayesian network is specified, i.e., its graph and numerical parameterization are given, the most common problem is *inference*. Namely, one is interested in the calculation of marginal or conditional probabilities of subsets of variables, such as the likelihood of observed evidence or the probability of some variables given evidence, or one is interested in identifying the most likely outcome of unobserved (hidden) variables given observed evidence. In the last decade, major progress has been accomplished in the theory of BNs. In particular, fast and exact inference algorithms has been developed when all the variables are discrete, all Gaussian or mixed discrete-Gaussian [8]. In this section, we present the JLO algorithm [22] as well as the David algorithm [11] which are the most popular algorithms for exact inference of discrete BNs. We recall here that the JLO and Dawid algorithms correspond respectively to the Forward-Backward and Viterbi algorithms when applied to the particular case of HMMs [31].

The JLO and Dawid algorithms proceeds in two steps. The first one consists in using graph-theoretic tools to transform the initial graphical structure of the BN into a specific graphical entity called the *junction tree*. In the second step, the junction tree is used as a channel to transmit and propagate the effect of observations (or evidence).

3.1. Construction of the junction tree

The first operation in the construction of the junction tree for BNs is the *moralization*. It consists in adding an extra undirected edge between any two nodes with a common child and subsequently removing directions. The undirected graph obtained this way is called the *moral* graph. The second operation consists in adding sufficient edges to the moral graph to make it *triangulated*. An undirected graph is triangulated (or chordal) if all cycles containing four or more nodes have a chord, i.e., an undirected edge between two non-consecutive nodes in the cycle. There are several

ways to add a chord in a cycle with length greater than four. Hence, the triangulation process is not unique. In general, it is desired to obtain a triangulation with a minimum number of additional edges. Unfortunately, the problem of automatically obtaining a minimal triangulation is NP-complete [36]. However, there exists some heuristic algorithms which work well in practice. For instance, the Maximum Cardinality Search Fill-In algorithm [32] can be used to obtain a triangulation of a given undirected graph. This algorithm can be implemented in linear time $O(N+l)$, where N is the number of nodes and l is the number of links in the graph. In the final operation, one identifies the set $\mathcal{C}$ of cliques[†] in the triangulated graph and forms a tree with these cliques in such way that resulting tree, the junction tree, satisfies the *running intersection property*. This property states that each variable which appears in two different cliques has to appear in all the cliques on the path between these two cliques. Figure 2 shows an example illustrating these different steps in the junction tree construction.

Attached with each edge linking two cliques C_1 and C_2 in the junction tree is a *separator* $S \stackrel{\triangle}{=} C_1 \cap C_2$. We denote the set of separators by $\mathcal{S}$. The main advantage of the junction tree representation is the fact that, as shown in [29], the joint probability distribution $P(X)$ can be factored as the product of clique marginals over separator marginals:

$$P(x) = \frac{\prod_{C \in \mathcal{C}} P(x_C)}{\prod_{S \in \mathcal{S}} P(x_S)} \tag{2}$$

where $P(x_C)$ and $P(x_S)$ are the marginal distributions over the variables in C and S respectively. Thus, probability calculations on X can be carried out locally and efficiently if the cliques are relatively small. The next subsection presents the message passing scheme of the JLO and Dawid algorithms leading to such local factorization of the JPD, in the light of observed evidence.

3.2. Propagation of evidence in the junction tree

Given the junction tree, the JPD $P(X)$ can be factored as

$$P(x) = \frac{\prod_{C \in \mathcal{C}} \phi_C(x_C)}{\prod_{S \in \mathcal{S}} \phi_S(x_S)} \tag{3}$$

where $\phi_C(x_C)$ (resp. $\phi_S(x_S)$) is a non-negative potential function on the clique $\mathcal{C}$ (resp. the separator $\mathcal{S}$). The collection of potentials $\Phi = \{\{\phi_C, C \in \mathcal{C}\}, \{\phi_\mathcal{S}, \mathcal{S} \in \mathcal{S}\}\}$ is termed a *representation* of $P(X)$. A factorizable distribution $P(X)$ may have many different representations, i.e., many collections of potentials which satisfy (3). For BNs, an *initial* representation is obtained from (1) in the following way. First, assign each X_i to just one clique. Second, for each clique C, define the potential ϕ_C to be either the product of $P(X_i|pa(X_i))$ over all X_i assigned to C, or 1 if no variable

[†]A clique is a subset of nodes which are fully connected and maximal, i.e, if a node is added to the subset, the latter does not remain fully connected.

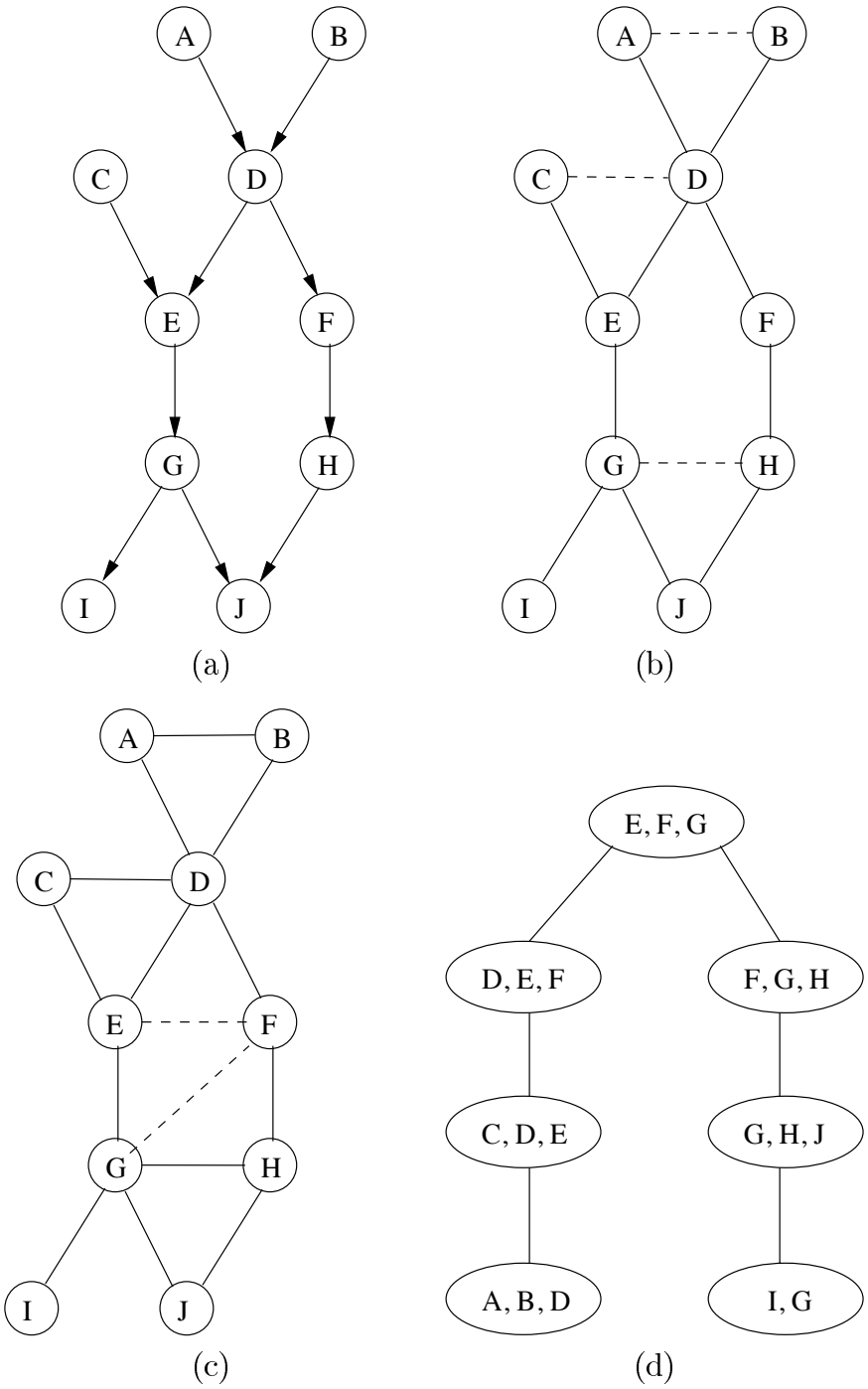

Figure 2: Illustration of the junction tree construction algorithm. (a) A directed acyclic graph. (b) Corresponding moral graph. (c) A triangulation of the moral graph. (d) A junction tree associated with the triangulated graph.

is assigned to C. Then, if ϕ_S is set to be 1 for each separator S, one obtains a representation of $P(X)$.

To propagate the effect of an observed evidence e, the JLO algorithm operates by transforming one representation to another, starting from the initial one modified by the incorporation of the evidence. The algorithm finishes with the *marginal* representation in which, for each clique C (resp. separator S), the potential ϕ_C (resp. ϕ_S) is equal to the marginal (joint) probability distribution for the variables in C (resp. S) and the evidence. The incorporation of evidence in the initial representation is done simply by setting $\phi_C(x_C)$ to 0 for any clique C containing an observed variable and for any configuration x_c involving a different state of the one observed. After this incorporation, the algorithm proceeds by passing a sequence of *flows* along the edges of the junction tree. Each flow from clique C_1 to C_2 updates the potentials of C_2 and the separator $S = C_1 \cap C_2$ in the following manner. Suppose that, prior to this flow, we have a representation Φ. Then, the activation of the flow yields a new representation Φ^* where the new potentials of C and S are[‡]

$$\phi^*_S = \sum_{C_1 \setminus S} \phi_{C_1} \quad ; \quad \phi^*_{C_2} = \frac{\phi^*_S}{\phi_S} \phi_{C_2} \tag{4}$$

and all the other potentials being unchanged. A *schedule* of such flows consists in updating all the cliques using the available information. This is done by choosing a clique C_r to be the *root* clique and, then, operating a recursive two-phase propagation scheme: *collecting* evidence and *distributing* evidence. In the collection phase, flows are activated along all the edges of the junction tree toward C_r. In the distribution phase, flows are activated out from C_r in the reverse direction. Once a schedule is complete, one obtains a new (final) representation Φ^f in which the potentials ϕ^f_C and ϕ^f_S of each clique C and separator S equal $P(x^h_C, e)$ and $P(x^h_S, e)$ respectively, where x^h_C (resp. x^h_S) is a configuration of the hidden variables in C (resp. S):

$$P(x) = \frac{\prod_{C \in \mathcal{C}} P(x^h_C, e)}{\prod_{S \in \mathcal{S}} P(x^h_S, e)} \tag{5}$$

Therefore, by marginalizing over the unobserved variables in any clique or separator, on gets the likelihood of observations $P(e)$. Also, by normalizing the potential at a clique C to sum 1, one get the posterior conditional probability $P(x^h_C | e)$ of the hidden variables in C given the evidence e. At this point, it is easy to note that the complexity of the JLO algorithm scales as the sum of the clique state-spaces[§].

The Dawid algorithm [11] is a slightly modified version of the JLO algorithm and allows the identification, with the same time complexity, of the most likely sequence of hidden states given observations [11]. There are only two modifications to operate (w.r.t. the JLO algorithm) and both are in the propagation scheme phase. The first one is to replace the sum-marginalization by a max-marginalization in the

[‡]The summation $\sum_{C_1 \setminus S}$ is over the state-space of variables that are in C_1 but not in S.

[§]A clique state-space is the product over each variable in the clique of the number of states of each variable

definition of a flow, i.e., to replace the summation by a maximization in (4). The second modification is in the distribution phase: once the potential of a clique is computed, one finds a configuration of its variables that maximizes the potential, this configuration is then considered as a new evidence when flows are activated. The running intersection property guarantees that when a variable outcome is fixed in one clique, that variable has the same outcome in all other cliques.

4. Multi-band speech recognition: the classical approach

The multi-band principle was originally motivated by an extensive research on the way humans process and recognize speech. This research, conducted by Harvey Fletcher [15] during the first half of the 20th century, suggests that the human auditory system recognizes speech using partial information across frequency, probably in the form of speech features that are localized in the frequency domain. However, Fletcher's work has been little known until 1994 when Jont B. Allen published a paper [1] in which he reviewed the work of Fletcher and also proposed to adapt a multi-band paradigm to automatic speech recognition. Many researchers have then studied this principle to build multi-band ASR systems [5, 14, 21, 27, 7].

When applied to automatic speech recognition, the multi-band principle can be viewed as a new architecture for ASR systems. In general, this architecture consists in dividing the frequency domain of the speech signal into several frequency sub-bands, then independent processing is applied in each sub-band. The application of such a principle generally leads to a multi-band ASR system which has the architecture represented in Figure 3.

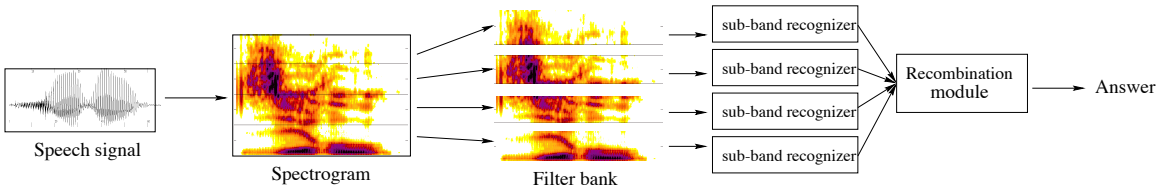

Figure 3: Classical multi-band architecture

In such a system, the speech signal is first passed to a filter-bank which splits it into several frequency bands. The signal in each band is then encoded into a stream of acoustic vectors, which are passed to a modeling or recognition stage. This stage is usually composed of Hidden Markov Models (HMM). During recognition, the HMMs scores are given to a recombination module, whose role is to deliver a unique answer to the recognition task. The inputs of the recombination module may either be the likelihoods that each HMM has generated the speech segment, or the label of the winning model in each band, or an ordered list of the concurrent HMMs.

Besides the motivation to mimic the human auditory system, this multi-band architecture has been also motivated by the following aspects. First, speech is characterized by *asynchrony* between frequency sub-bands, in the sense that stationary segments transition may occur at different times in the time-frequency domain. Thus, asynchrony may be taken into account by local frequency processing, this in

turn could lead to higher fidelity speech modeling than traditional HMM modeling. Indeed, in the latter, speech segments are implicitly assumed to contain phoneme-synchronous information given that features extraction uses the whole frequency band. Second, the information contained in some sub-bands may be more relevant than in the other sub-bands. Thus, an "appropriate" weighting of each sub-band could improve recognition accuracy. Finally, this multi-band architecture could improve the recognition robustness to band-limited noise w.r.t. standard HMM-based systems. Indeed, even when the noise covers only one frequency sub-band, the latter would yield bad performances since the acoustic features (MFCC coefficients in general) are calculated on the whole spectrum and are then all corrupted. Using this architecture, only the acoustic features corresponding to the noisy sub-band would be corrupted. One can then exploit the non-corrupted information in the other sub-bands for recognition.

While the ideas leading to multi-band speech recognition are attractive, the classical architecture described above has many drawbacks however. For instance, the sub-bands are assumed mutually independent which is an unrealistic hypothesis. Moreover, the information contained in one sub-band is not discriminative in general. In addition, it is not easy to deal with asynchrony, particularly in continuous speech recognition. As a consequence, the recombination step can be a very difficult task.

In the next section, we present a new approach for multi-band speech recognition which has the advantage to overcome *all* the limitations (mentioned above) of the classical multi-band systems.

5. Multi-band speech recognition: the DBNs perspective

Let us assume that we are given a vocabulary V of $|V|$ words. The basic idea behind our approach is the following: for each word $v \in V$, instead of considering an independent HMM for each sub-band (as in the classical multi-band approach), we build a more complex but uniform DBN on the time-frequency domain by "coupling" all the HMMs associated with the different sub-bands. By coupling we mean adding (directed) links between the variables in order to capture the dependency between sub-bands. A natural question is: what are the "appropriate" links to add? Probably the best answer is to learn the graphical structure (i.e., the dependencies between variables) from data. However this strategy, known as *structural learning* [19], which is extremely interesting and which we are currently investigating [12, 13] is beyond the scope of this paper. Instead, our philosophy in this paper is to (first) impose a "reasonable" graphical structure (for all words) and then see whether the principle of our new multi-band approach is promising. If yes, this "reasonable" structure could be used as *prior knowledge* [20] in a structural learning procedure.

5.1. Model definition

We build such "reasonable" structure using the following computational and physical criteria. We want a model where no continuous variable has discrete children in order to apply an exact inference algorithm. Indeed, only approximate inference is possible in networks where continuous variables have discrete children[30]. We

also want a model with a relatively small number of parameters and for which the (exact) inference algorithms are tractable. Finally, we want to have links between the hidden variables along the frequency axis in order to capture the asynchrony between sub-bands. A simple model which satisfies these criteria is shown in Figure 4. In this BN, the hidden variables of sub-band n are linked to those of sub-band $n+1$ in such way that the state of a hidden variable in sub-band $n+1$ at time t is conditioned by the state of two hidden variables: at time $t-1$ in the same sub-band and at time t in sub-band n. Each $H_t^{(n)}\left(=H_t^{(n)}(v)\right)$ is a discrete variable

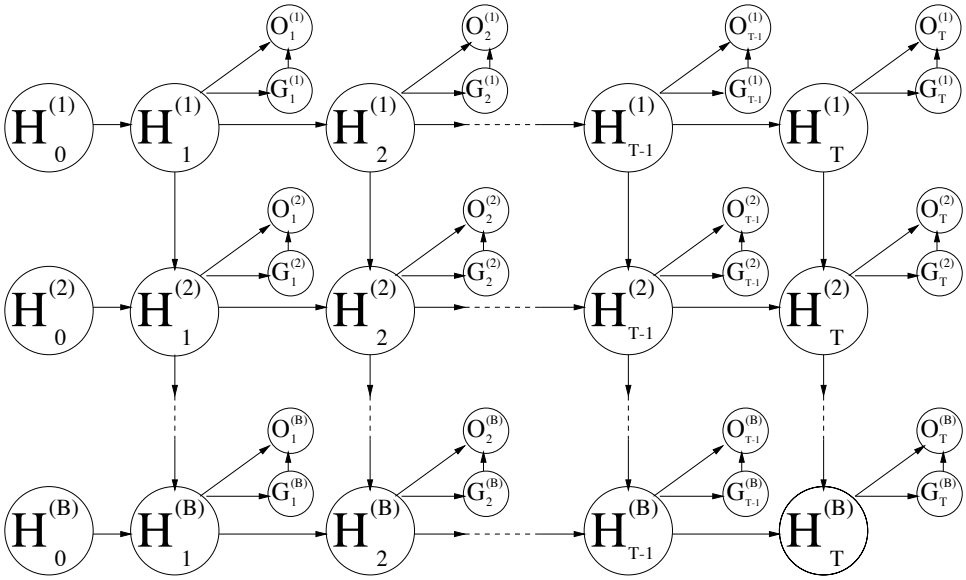

Figure 4: B-band dynamic Bayesian network

taking its values in the set of ordered labels $I_v = \{1_v, ..., m_v\}$, $|I_v|$ is the number of hidden states. Each $O_t^{(n)}\left(=O_t^{(n)}(v)\right)$ is a continuous variable with a Gaussian-mixture distribution (given an outcome of the corresponding hidden variable $H_t^{(n)}$) representing the observation vector at time t in sub-band n ($n = 1, ..., B$), B is the number of sub-bands. Each $G_t^{(n)}\left(=G_t^{(n)}(v)\right)$ is a discrete variable taking its values in the set $J = \{1, ..., M\}$, M is the number of Gaussian components in each mixture¶. We impose a left-to-right topology on each sub-band and assume that the model parameters are stationary. Therefore, given a word $v \in V$ (and for each $(i, j, k, p) \in I_v^3 \times J$) , the numerical parameterization Θ_v of its B-band DBN model is:

$$\begin{cases} a_{ij}(v) \triangleq P(H_t^{(1)}(v) = j | H_{t-1}^{(1)}(v) = i) \\ u_{ijk}^{(n)}(v) \triangleq P(H_t^{(n)}(v) = k | H_t^{(n-1)}(v) = i, H_{t-1}^{(n)}(v) = j) \quad \text{for} \quad n = 2, ..., B \\ w_{ip}^{(n)}(v) \triangleq P(G_t^{(n)}(v) = p | H_t^{(n)}(v) = i) \quad \text{for} \quad n = 1, ..., B \\ b_{i,p}^{(n)}(v, \cdot) \triangleq P(O_t^{(n)}(v) = \cdot | H_t^{(n)}(v) = i, G_t^{(n)}(v) = p) \quad \text{for} \quad n = 1, ..., B \end{cases} \tag{6}$$

¶The use of the variables $G_t^{(n)}$ is not necessary to define the model. We use them only to have a consistency with the fact that exact inference is possible in mixed discrete-continuous BNs only if the continuous variables are Gaussian [8].

where $b_{i,p}^{(n)}(v,\cdot)$ is a Gaussian with mean $\mu_{i,p}^{(n)}(v)$ and covariance $\Sigma_{i,p}^{(n)}(v)$. The asynchrony between sub-bands is taken into account by allowing all the $u_{ijk}^{(n)}(v)$ to be non-zero, except when $k < j$ or $k > j+1$ because of the left-to-right topology.

Note that our model is a mixed discrete-Gaussian BN. Therefore, in principle, inference should be done using the Lauritzen algorithm [30]. However, in our setting, inference will always involve a *complete* observations set of the continuous variables. In other words, *all* the $O_t^{(n)}$ are observed when our model is infered. Thus, even tough the B-band DBN is mixed discrete-Gaussian, the JLO and Dawid algorithms (which apply to discrete networks) are enough to perform inference in this setting. Note also that our model is a special case of the so called *tree structured HMMs* [16]. However, we do not use the variational approach described in [16] to infer this model because we are interested in exact inference.

We now stretch the advantages of such approach to multi-band ASR and describe briefly some related work. Unlike HMMs, our multi-band DBN provides *a* modeling of the frequency dynamics of speech. Unlike to the classical multi-band approach, our DBN allows interaction between sub-bands and the possible asynchrony between them easily handeled. Moreover, our model uses the information contained in all sub-bands and no recombination step is needed. A related work has been proposed in [18, 17] where a multi-band Markov random field is analyzed by mean of Gibbs distributions. This approach (unlike ours) does not lead however to exact nor fast inference algorithms and assumes a linear model for asynchrony between sub-bands. In our approach, the asynchrony is learned from data. In term of introducing frequency dynamics in the modeling process, a related work has been proposed in [35, 34, 33]. In this work, the authors propose a new model, called HMM2, which is an HMM "mixture" consisting in a primary (classical) HMM, modeling the temporal properties of the speech signal, and a secondary HMM modeling its frequency properties. This secondary HMM is inserted at the level of each state of the primary HMM, estimating local emission probabilities of acoustic feature vectors (consisting in spectral features). Consequently, the components of an acoustic vector are assumed to be generated by the secondary HMM, the goal being to perform a (state dependent) dynamical spectral warping to complement the (classical) time warping done by the primary HMM. This HMM2 has then been used as a decoder as well as feature extractor and tested in various conditions. In the case of clean speech, performances were showed to be comparable to classical MFCC-based HMM systems. However, in the case of noisy speech, the performances are so far still limited by the choice of the spectral features, which are less robust to noise than MFCCs. Besides the introduction of some modeling of the frequency dynamics of speech, HMM2 is different from our model in all aspects: topology and parameterization. Indeed, our model is a "pure" multi-band model in the sense that the frequency axis is divided in a static way, we use MFCCs in the parameterization and obtain some good performances in the presence of noise as compared to classical HMM systems (as we will see later).

5.2. Construction of the junction tree

Now that the graphical structure and the numerical parametrization of our multi-band DBN are specified, inference can be performed using the JLO or/and the Dawid algorithms. The only step remaining is to associate a "minimal" junction tree to our network, in the sense that no other junction tree can lead to faster inference. This is of particular importance given that the decoding efficiency requires a junction tree with "small" clique state-spaces. As explained in section 3, the first step in this construction is the moral graph. Figure 5 displays the moral graph of our multi-band BN. In the one-band (i.e. HMMs) or two-band cases, finding a minimal junction tree is obvious [9] because the moral graph is triangulated as it is (consider $B = 1$ or $B = 2$ in Figure 5). This is not true any more when $B > 2$. Since the problem of automatically finding minimal junction trees for arbitrary BNs is NP-complete [36], we need to find an appropriate (analytical) technique to derive a minimal junction tree for our particular B-band BN. We do so as follows:

First, it is clear from the moral graph that the only cliques which contain the variables $O_t^{(n)}$ and $G_t^{(n)}$ are of the form $H_t^{(n)}G_t^{(n)}O_t^{(n)}$. For the remaining cliques (which all contain only the hidden variables $H_t^{(n)}$), we proceed by induction. If $B = 1$, it is obvious that the clique which has to be linked to $H_t^{(1)}G_t^{(1)}O_t^{(1)}$ is $H_t^{(1)}H_{t-1}^{(1)}$. If $B = 2$ it is easy to check from the moral graph (which is triangulated as it is) that the clique which has to be linked to $H_t^{(1)}G_t^{(1)}O_t^{(1)}$ (resp. $H_t^{(2)}G_t^{(2)}O_t^{(2)}$) is $H_t^{(1)}H_{t-1}^{(1)}H_{t-1}^{(2)}$ (resp. $H_t^{(1)}H_t^{(2)}H_{t-1}^{(2)}$). Then, by induction one can prove that the clique which has to be linked to $H_t^{(n)}G_t^{(n)}O_t^{(n)}$ is $H_t^{(1)}\cdots H_t^{(n)}H_{t-1}^{(n)}\cdots H_{t-1}^{(B)}$. The time slices $t = 1$ and $t = T$ are then treated separately to remove the variables which are not necessary to satisfy the runing intersection property. The resulting junction tree is shown in Figure 6. We thus have a computationally optimal tree to propagate the effect of observed evidence.

The complexity of the JLO and Dawid algorithms scales as the sum of clique state-spaces. Therefore, given the asynchrony assumptions, the left-to-right topology and our junction tree, the total complexity to infer this B-band DBN is $O(MBD|I_v|^BT)$ where D is the dimension of the acoustic vectors.

5.3. Model parameters estimation

So far, we have assumed that parameters of the B-band DBN are known for each word. In this section, we present an algorithm of parameters estimation. In all the experiments we carry out later, we learn the model of each word independently of the others, i.e., we do not perform embedded training. We emphasize however that this is not a constraint of our system. Indeed, embedded training can be performed exactly as in the HMMs setting given that all graphical structures of the models are the same. Also, the use of words as acoustic units in not a constraint neither, any kind of acoustic units (phonemes, diphones...) can be used without any particular change in our methodology.

In order to simplify the notation in the formulae below, we drop the reference to the word under consideration. Suppose that we have (for a given word v) an

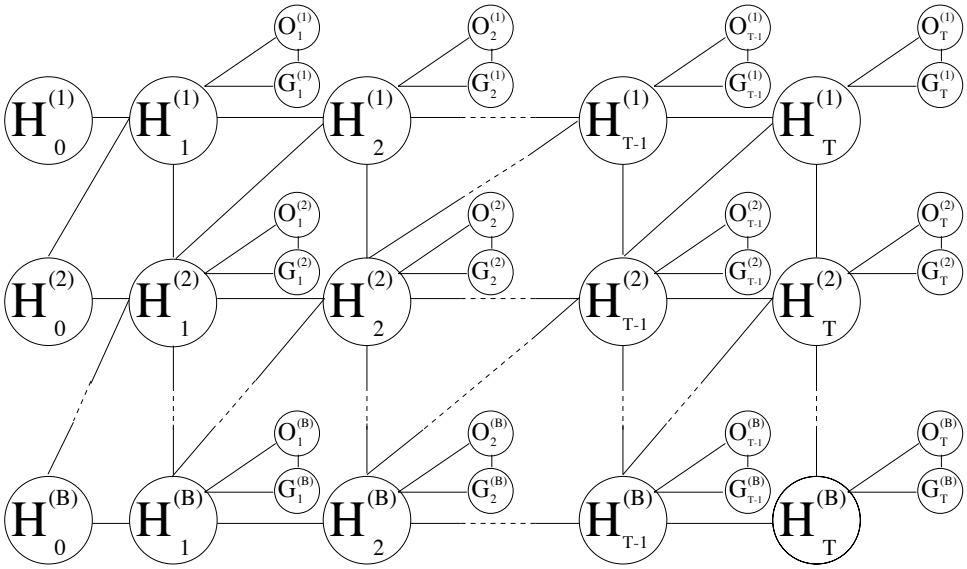

Figure 5: Moral graph of the B-band Bayesian network.

observation vector $o = (o_1^{(1)}, ..., o_T^{(1)}, ..., o_1^{(B)}, ..., o_T^{(B)})$. Let

$$\mathcal{H} = \left\{ h = (h_0^{(1)}, ..., h_T^{(1)}, ..., h_0^{(B)}, ..., h_T^{(B)}) : h_t^{(n)} \in \{1, ..., m\} \right\}$$

be the set of all possible trajectories of the hidden process defined by the $H_t^{(n)}$. Then, from (1) and by marginalization over $\mathcal{H}$, the likelihood is given by:

$$P(o) = \sum_{h \in \mathcal{H}} \left\{ \prod_{t=1}^{T} a_{h_{t-1}^{(1)} h_t^{(1)}} \prod_{n=2}^{B} u_{h_t^{(n-1)} h_{t-1}^{(n)} h_t^{(n)}} \right\} \left\{ \prod_{t=1}^{T} \prod_{n=1}^{B} \left(\sum_{p=1}^{M} w_{h_t^{(n)} p} b_{h_t^{(n)} p}^{(n)}(o_t^{(n)}) \right) \right\}.$$

Therefore, by aplying the EM algorithm, the auxiliary function can be decomposed as the sum of terms depending each on one component of the parameters set. Thus, solving the parameters estimation problem comes back to a simple generalization of the Baum-Welch algorithm. This is made possible essentially because we have imposed that continuous variables are conditioned by discrete ones. The re-estimation formulae are obtained as follows: suppose that we have estimated the parameters at iteration l and define for $(i, j, k, p) \in \{1, ..., m\}^3 \times \{1, ..., M\}$ (here we assume that the number of hidden states is the same for all words and equals some integer m, i.e., $|I_v| = m, \forall v$):

$$\begin{cases} \psi_t^{(1)}(i,j) \triangleq P(H_{t-1}^{(1)} = i, H_t^{(1)} = j | o) \\ \psi_t^{(n)}(i,j,k) \triangleq P(H_t^{(n-1)} = i, H_{t-1}^{(n)} = j, H_t^{(n)} = k | o) \quad \text{for} \quad n = 2, ..., B \\ \psi^{(1)}(i,j) \triangleq \sum_{t=1}^{T} \psi_t^{(1)}(i,j) \\ \psi^{(n)}(i,j,k) \triangleq \sum_{t=1}^{T} \psi_t^{(n)}(i,j,k) \quad \text{for} \quad n = 2, ..., B \\ \phi_t^{(n)}(i,p) \triangleq P(H_t^{(n)} = i, G_t^{(n)} = p | o) \quad \text{for} \quad n = 1, ..., B \\ \phi^{(n)}(i,p) \triangleq \sum_{t=1}^{T} \phi_t^{(n)}(i,p) \quad \text{for} \quad n = 1, ..., B. \end{cases} \tag{7}$$

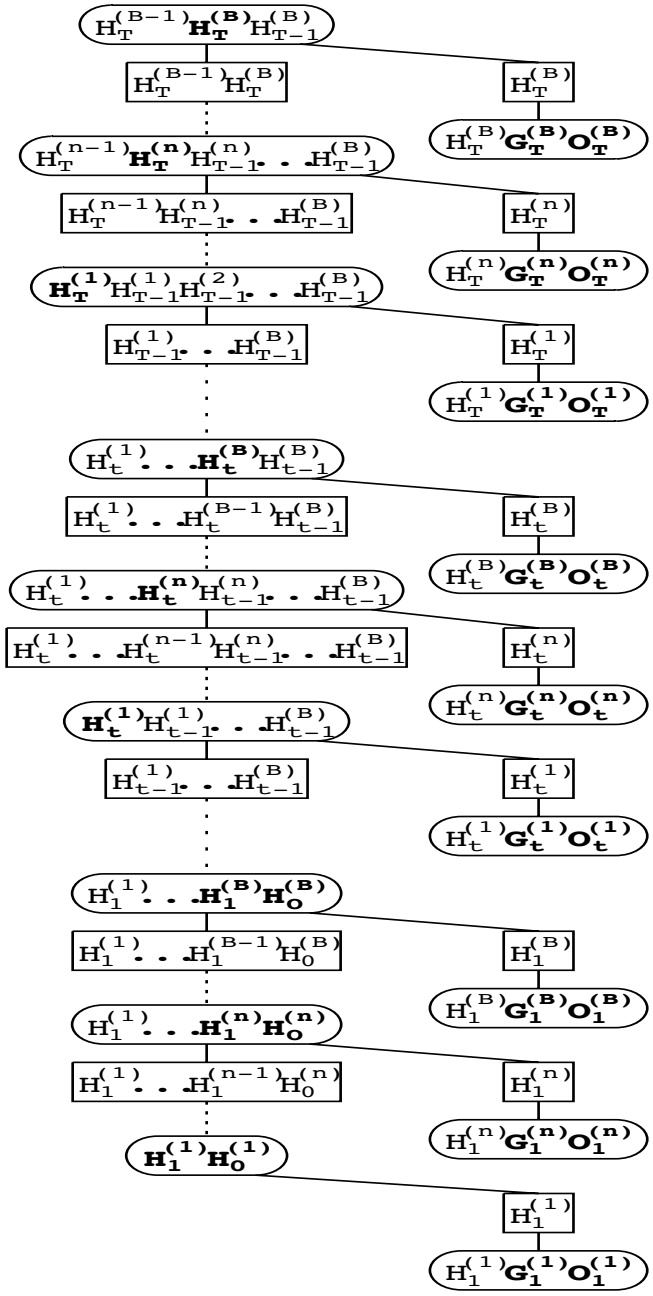

Figure 6: Junction tree of the B-band Bayesian network. Cliques and separators are respectively represented by ellipsoids and rectangles. For each clique, the variables in bold are those which are *assigned* to that clique in order to obtain an *initial representation* of the JPD (see section 3).

Then, the new parameters at iteration $l+1$ are given by[||]

$$a_{ij} = \frac{\psi^{(1)}(i,j)}{\sum_{k=1}^{m} \psi^{(1)}(i,k)}$$

[||]For sake of notational simplicity, we drop the iteration index.

$$u_{ijk}^{(n)} = \frac{\psi^{(n)}(i,j,k)}{\sum_{l=1}^{m} \psi^{(n)}(i,j,l)}$$

$$w_{ip}^{(n)} = \frac{\phi^{(n)}(i,p)}{\sum_{q=1}^{M} \phi^{(n)}(i,q)}$$

$$\mu_{i,p}^{(n)} = \frac{\sum_{t=1}^{T} \phi_t^{(n)}(i,p) o_t^{(n)}}{\phi^{(n)}(i,p)}$$

$$\Sigma_{i,p}^{(n)} = \frac{\sum_{t=1}^{T} \phi_t^{(n)}(i,p)(o_t^{(n)} - \mu_{i,p}^{(n)})(o_t^{(n)} - \mu_{i,p}^{(n)})^*}{\phi^{(n)}(i,p)}$$

What remains is to efficiently compute the posterior probabilities $\psi_t^{(1)}(i,j)$, $\psi_t^{(n)}(i,j,k)$ and $\phi_t^{(n)}(i,p)$ defined in equation (7). All these posterior probabilities can be efficiently computed using the JLO algorithm which allows the computation of marginal and conditional probabilities of clique variables. We point out also that in the full-band case ($B = 1$) which collaps to an HMM, the computation of $\psi_t^{(1)}(i,j)$ and $\phi_t^{(1)}(i,p)$ using the JLO algorithm is (exactly) equivalent to the Forward-Backward algorithm.

6. Application to isolated speech recognition

For an isolated speech recognition task, the decoding algorithm is readily given by the material described in section 3. Indeed, once we have learned a B-band DBN model Θ_v for each $v \in V$, then given a speaker utterance o, we compute the likelihood $P(o|\Theta_v)$ for each $v \in V$ using the JLO algorithm and choose the word v^* such that

$$v^* = \text{argmax}_v \; P(o|\Theta_v)$$

to be the pronounced word. The computational complexity of this decoding algorithm is $O(MBm^BT)$, we recall that M is the number of Gaussian components in each mixture and m is the number of hidden states.

Experiments

We now evaluate the performance of the B-band DBN on an isolated digit recognition task. We compare our model to HMMs, a classical multi-band (CMB) system and a synchronous "multi-band" Bayesian network. The experiments are carried out on the isolated part of the Tidigits database** in which 112 (resp. 113) speakers are used for training (resp. test). Each speaker utters 11 digits twice. The parameterization for the classical full-band HMM is done as follows: 25ms frames with a

**We emphasize here that our purpose in this paper is not to perform benchmark tests, i.e., our goal here is not to tune the parameters in order to achieve the highest performances. Rather, we provide comparisons using baseline systems in order to *illustrate* the capabilities of each system and have a fair judgment on their potentail.

frame shift of 10ms, each frame is passed through a set of 24 triangular filters resulting in a vector of 35 features, namely, 11 static MFCC (the energy is dropped), 12 Δ and 12 $\Delta\Delta$. For our model, we present experiments in the case of 2, 3 and 4-band BN. The parameterization for the 2-band DBN is done as follows: each frame is passed through the 14 first (resp. last 10) filters resulting in the acoustic vector of sub-band 1 (resp. sub-band 2). Each vector contains 17 features: 5 static MFCC, 6 Δ and 6 $\Delta\Delta$. The resulting bandwidths of sub-bands 1 and 2 are $[0..1467Hz]$ and $[1211Hz..10000Hz]$ respectively. For the 3-band BN, each frame is passed through the first 8, second 8 and last 8 filters resulting in the acoustic vector of sub-band 1, 2 and 3 respectively. Each vector contains 11 features: 3 static MFCC, 4 Δ and 4 $\Delta\Delta$. The resulting bandwidths of sub-bands 1, 2 and 3 are $[0..692Hz]$, $[615Hz..2152Hz]$ and $[1777Hz..10000Hz]$ respectively. Similarly, for the 4-band BN, each frame is passed through the first 6, second 6, third 6 and last 6 filters. Each resulting vector contains 8 features. The resulting bandwidths of sub-bands 1, 2, 3 and 4 are $[0..538Hz]$, $[461Hz..1000Hz]$, $[923Hz..3158Hz]$ and $[2607Hz..10000Hz]$ respectively. In all the experiments, for every digit and all models, the number of hidden states is six ($m = 6$) and we have a single Gaussian per state with a diagonal covariance matrix. Table 1 shows the recognition scores obtained using the 2, 3 and 4-band BNs and also the score with a classical full-band HMM. In this experiment, both train and test are on clean speech. In these results the three B-band BNs all

Model	HMM	2-band	3-band	4-band
Score	93.4%	**97.4%**	**97.3%**	**95.4%**

Table 1: Recognition scores of the HMM and the B-band DBN ($B = 2, 3, 4$) on clean speech.

outperform the HMM recognizer that we tested. We conclude that taking into account the frequency dynamics leads to a higher fidelity speech modeling. One can notice however (in this experiment) that when the number of sub-bands increases the accuracy decreases. This should not be understood as a characteristic of our multi-band system. Probably one explanation is the fact that we are using the same amount of data to estimate models with an increasing number of parameters. We believe however that this behavior is mainly due to the ad-hoc choice of sub-bands bandwidths. For instance, sub-band n is more relevant than sub-band n' in the B-band DBN if $n' > n$, in the sense that it governs the behaviour of sub-band n'. Thus, for example, sub-band 1 is more relevant than all the others and in the parameterization we chose, when the number of sub-bands increases the amount of information contained in sub-band 1 decreases. The optimization of the sub-bands bandwidths is not our major concern in this work because we do not perform benchmark tests. What should be retained from these results is that, even with such ad-hoc choice of sub-bands frequencies, the B-band DBN outperform HMMs. This is in fact a major advantage since, to the best of our knowledge, the only multi-band systems which out-perform HMMs in clean conditions use the full-band parameterization as an ad-

ditional "sub-band". Such a manipulation is conceptually unrealistic in our opinion and penalizes the systems in noisy conditions.

In the following experiments, training is done on clean speech and test is done on noisy speech. We show the performances of our model when the noise is additive and corrupt one spectral sub-band with two different bandwidths. We compare a 2-band DBN to HMMs and two other models. The first one is a standard 2-band model where the recombination is performed by a multi-layer perceptron (MLP), we term this model CMB-MLP. The topology of this MLP is: 22 input, 15 hidden and 11 output neurons. In the second model that we term Sync, for each frame, we concatenate the acoustic vectors of sub-band 1 and 2 and use the resulting vector (34 features) as an input for the HMM-based system. It is important to note that, since we use diagonal covariances, Sync is equivalent to a 2-band DBN where a complete synchrony is imposed between the two bands. Indeed, given that covariances are diagonal, the HMM representing Sync can be viewed as the DBN shown in Figure 7, where $B = 2$ and each variable H_t (resp. G_t) takes its values in the set $\{1, ..., m\}$ (resp.$\{1, ..., M\}$). The model of Figure 7 corresponds in turn to a completely synchronous B-band DBN. Therefore, the comparison between Sync and our 2-band DBN will be a good indication about the importance of asynchrony. The issue of asynchrony in multi-band ASR has been studied by many researchers. For instance, in [24] the authors conclude that considering asynchrony in multi-band ASR may improve the acoustic modeling. They failed however, in [25], to improve the recognition performances when relaxing the synchrony constraints in a multi-band system, they then conclude that asynchrony is not advantageous. We believe that if this argument is true, it is only in the sense of incorporating asynchrony assumptions in a *classical* multi-band system. In other words, even if this argument is true, it does not mean that asynchrony does not exist or is irrelevant. It only means that it is difficult to exploit and deal with asynchrony in classical multi-band systems. As we will see below, The DBNs perspective to multi-band ASR suggests that asynchrony is in fact advantageous.

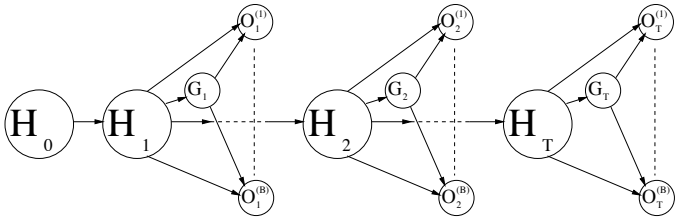

Figure 7: Synchronous B-band dynamic Bayesian network

The noisy speech (in test) is obtained by adding to the clean one, at different SNRs, band-pass filtered white noises with different bandwidths: $[5000Hz..10000Hz]$ for Noise A and $[2000Hz..7000Hz]$ for Noise B, the SNR being estimated as:

$$SNR = 10 \log_{10} \left(\frac{\text{Signal Energy}}{\text{Noise Energy}} \right).$$

Therefore, in both cases, only sub-band 2 is corrupted [††]. Table 2 (resp. Table 3) gives the recognition rates obtained for Noise A (resp. Noise B) using the four models (HMM, Sync, CMB-MLP and the 2-band DBN). For both noises, the 2-band DBN largely outperforms all the other models. Even when the scores of the latter are extremely low (sometimes close to random decision), our model still yield relatively good recognition rates. This indicates that our model is well adapted to the case of band-limited noisy speech. It is also remarkable how huge are the differences between the scores of our model and those of Sync. We believe this illustrates the importance of asynchrony in multi-band speech modeling.

Noise A: 5-10 KHz	HMM	Sync	CMB-MLP	2-band DBN
SNR 26db	53.4%	43.5%	59.5%	**84.9**%
SNR 20db	38.8%	27.2%	39.0%	**77.9**%
SNR 14db	26.3%	18.8%	23.4%	**70.8**%
SNR 8db	18.4%	11.6%	14.7%	**65.5**%
SNR 2db	13.7%	9.3%	10.4%	**63.2**%

Table 2: Recognition scores for Noise A using the different models

Noise B: 2-7 KHz	HMM	Sync	CMB-MLP	2-band DBN
SNR 26db	52.3%	45.8%	54.0%	**82.7**%
SNR 20db	46.2%	32.1%	42.8%	**71.2**%
SNR 14db	39.0%	20.1%	37.3%	**60.8**%
SNR 8db	32.5%	11.3%	33.7%	**54.4**%
SNR 2db	28.6%	9.5%	26.9%	**49.4**%

Table 3: Recognition scores for Noise B using the different models

7. Application to continuous speech recognition

In a continuous speech recognition task, given a B-band DBN model of each word in the vocabulary and a speaker utterance, the goal is to identify the most likely sequence of words given the observation. A naive solution would be to use a B-dimensional Viterbi algorithm [26] which is computationally very expensive. In this section, we present an efficient decoding algorithm which relies essentially on a state-augmented B-band DBN model, we then show experiments on a connected digits recognition task.

7.1. Decoding algorithm

The basic idea is to build a new B-band DBN model which represents all the words in the vocabulary, decoding is then performed by inferring this new DBN. Precisely,

[††]Obviously, in most real world applications one does not know a priori which sub-bands are corrupted, these has to be detected using some noise estimation algorithm.

the graphical structure of this new B-band DBN is the same as the one of Figure 3, the difference is that the variables do not depend any more on the word under consideration, and each variable $H_t^{(n)}$ takes now its values in the set $I = \bigcup_{v \in V} I_v$. To complete the definition of this new DBN we need to specify the conditional probabilities of the hidden and the observed variables. Let $(i, j, k, p) \in I^3 \times J$ such that $(i, j, k) \in {I_v}^3$ for some $v \in V$. Then, the observation's conditional probabilities are simply given by those corresponding to each word, namely:

$$P(O_t^{(n)} = \cdot | H_t^{(n)} = i, G_t^{(n)} = p) \triangleq b_{i,p}^{(n)}(v, \cdot).$$

To specify the hidden process parameterization we need to include the language model, we also make some (a)synchrony assumptions: we still allow complete asynchrony *inside* a word, but we impose a full synchrony of all sub-bands when *transiting* between words. Precisely, since we have a left-to-right topology, the only non-zero conditional probabilities are the following:

• The synchronous transition between two (not necessarily different) words v and v':

$$P(H_t^{(1)} = 1_v | H_{t-1}^{(1)} = m_{v'}) \triangleq P(v|v')$$

$$P(H_t^{(n)} = 1_v | H_t^{(n-1)} = 1_v, H_{t-1}^{(n)} = m_{v'}) \triangleq P(v|v')$$

where $P(v|v')$ is given by the language model.

• The inside-word conditional probabilities:

$$P(H_t^{(1)} = j | H_{t-1}^{(1)} = i) \triangleq a_{ij}(v)$$

$$P(H_t^{(n)} = k | H_t^{(n-1)} = i, H_{t-1}^{(n)} = j) \triangleq u_{ijk}^{(n)}(v).$$

Now we have a completely defined B-band model on which decoding can be performed. To do so, we use the Dawid algorithm [11] which allows the identification (with the same time complexity as the JLO algorithm [22]) of the most likely sequence of hidden states given observations.

Given the (a)synchrony assumptions and the left-to-right topology, the total complexity of this decoding algorithm is $O(MBm^B T + |V|^2 T)$. We point out also that in the 1-band case (i.e. HMMs), this algorithm is equivalent to Viterbi decoding.

7.2. Experiments

The experiments are carried out on the connected part of the Tidigits database in which 112 (resp. 113) speakers are used for training (resp. test). Each speaker utters 77 sentences resulting in 8642 sentences for training and 8701 for test, each sentence contains between 1 and 7 digits. We show comparisons[‡‡] of the performances of a

[‡‡]At the time of writing we do not have a continuous version of the CMB system to show its performances. However, we expect our system to have even better performances than such a system as compared to the results obtained in the isolated task. Indeed, the latter is the ideal setting for a CMB system because there is no word-asynchrony to deal with, and it is well known that recombination is a more difficult task in the continuous setting than in the isolated one. Our system does not have such discrepancy. Also, we do not show the results of the Sync model because it always yields the lowest performances.

2-band DBN with a single Gaussian per state to HMMs with a different number of Gaussian components in each mixture. For every digit and the silence model, the number of hidden states is seven ($m = 7$) and all the covariance matrices are diagonal. We use a uniform language model, i.e., $P(v|v') = \frac{1}{12}$ (eleven digits + silence). The parameterization of the classical full-band HMM is done as in the isolated task (see the previous section). The parameterization of the 2-band DBN is done as follows: each frame is passed through the 16 first (resp. last 8) filters resulting in the acoustic vector of sub-band 1 (resp. sub-band 2). Each vector contains 17 features: 5 static MFCC, 6 Δ and 6 $\Delta\Delta$. The resulting bandwidths of sub-bands 1 and 2 are $[0..2152Hz]$ and $[1777Hz..10000Hz]$ respectively. The training of all models is done on clean speech only. The test however is performed on noisy speech which is obtained by adding, at different SNRs, a band-pass filtered white noise with a bandwidth of $[3000Hz..6000Hz]$. Table 4 and 5 show respectively the digit and phrase accuracy that we obtain using both models. If one compares the 2-band DBN with HMM-1G which both have a single Gaussian per state, one sees that our model largely outperforms the HMM-1G model. One can argue that this may be due to the fact that our model uses (slightly) more parameters than the other model. The comparison between our model and the other HMMs (which have more than 2 Gaussian components per state) shows the opposite of this argument. Indeed, all these HMMs use much more parameters than the 2-band DBN, still our model yield the best performances. Particularly, the more the SNR is low the higher is the accuracy of the 2-band DBN as compared to the HMMs. This illustrates the potential of our approach in exploiting the information contained in the non-corrupted sub-band. In summary, the behavior of our system in the continuous task remains consistent as compared to the isolated task and its performances on this illustrative experiment are impressive. This shows that the DBNs perspective to multi-band speech recognition is very promising.

Noise 3-6 KHz	HMM-1G	HMM-2G	HMM-4G	HMM-8G	2-band DBN (1G)
SNR 26 db	89.95%	92.69%	97.20%	96.82%	**96.16**%
SNR 20 db	82.17%	85.17%	94.19%	93.59%	**94.89**%
SNR 14 db	73.27%	75.33%	87.44%	86.64%	**90.81**%
SNR 8 db	62.57%	59.57%	73.85%	72.91%	**82.27**%
SNR 2 db	58.86%	40.82%	54.60%	53.48%	**75.51**%

Table 4: Digit accuracy rates(nG means n Gaussian components per state)

Noise 3-6 KHz	HMM-1G	HMM-2G	HMM-4G	HMM-8G	2-band DBN (1G)
SNR 26 db	71.47%	79.05%	92.00%	90.77%	**89.42**%
SNR 20 db	52.49%	59.38%	84.09%	82.65%	**85.90**%
SNR 14 db	35.69%	40.13%	69.22%	67.29%	**74.67**%
SNR 8 db	20.90%	20.55%	46.00%	43.17%	**53.86**%
SNR 2 db	10.97%	9.696%	23.82%	22.52%	**39.87**%

Table 5: Phrase accuracy rates (nG means n Gaussian components per state)

8. Conclusion

We presented a new approach to multi-band speech recognition which consists in building a dynamic Bayesian network to model speech in the time-frequency domain. We developed all the necessary algorithmic material to apply it in isolated and continuous speech recognition. The experiments we carried out illustrate the potential of this new approach as compared to classical HMMs and classical multi-band systems. Generally speaking, this paper shows that the DBNs perspective is a very promising framework in the field of multi-band and noisy speech recognition. We emphasize here that the approach we presented in this paper can be further improved either by applying a structural learning procedure to learn the speech multi-band model from data, or by considering more complex dependencies between the variables of the hidden process in the B-band DBN with *no* additional cost in the inference complexity. Indeed, one can consider the more complex model shown in Figure 8 for a single time slice. This DBN has the same junction tree as the DBN we considered in this paper. Thus, it has the same inference complexity as our B-band DBN if the latter is assumed ergodic (i.e., no left-to-right topology is assumed). Moreover, such model would illustrate the importance of cross-correlation between sub-bands. This, as well as benchmark experiments, will be the purpose of future works.

References

[1] J. Allen. How do humans process and recognize speech. *IEEE Trans. Speech and Audio Processing*, 2(4):567–576, 1994.

[2] J. A. Bilmes. Data-driven extensions to hmm statistical dependencies. In *International Conference on Spoken Language Processing (ICSLP)*, 1998.

[3] J. A. Bilmes. Buried markov models for speech recognition. In *International Conference on Acoustic, Speech and Signal Processing (ICASSP)*, 1999.

[4] J. A. Bilmes. *Natural Statistical Models for Automatic Speech Recognition*. PhD thesis, International Compute Science Institute, Berkeley, California, 1999.

[5] H. Bourlard and S. Dupont. A new ASR approach based on independent processing and recombination of partial frequency bands. International Conference on Spoken Language Processing (ICSLP), 1996.

[6] E. Castillo, J.M. Gutirrez, and A.S. Hadi. *Expert Systems And Probabilistic Network Models*. Springer-Verlag, New York, 1997.

[7] C. Cerisara and D. Fohr. Multi-band automatic speech recognition. *Computer Speech and Language*, 15(2):151–174, 2001.

[8] R. Cowell, A. Dawid, S. Lauritzen, and D. Spiegelhalter. Probabilistic Networks and Expert Systems. *Springer-Verlag*, 1999.

[9] K. Daoudi, D. Fohr, and C. Antoine. A new approach for multi-band speech recognition based on probabilistic graphical models. In *International Conference on Spoken Language Processing (ICSLP), Beijing, China*, October 2000.

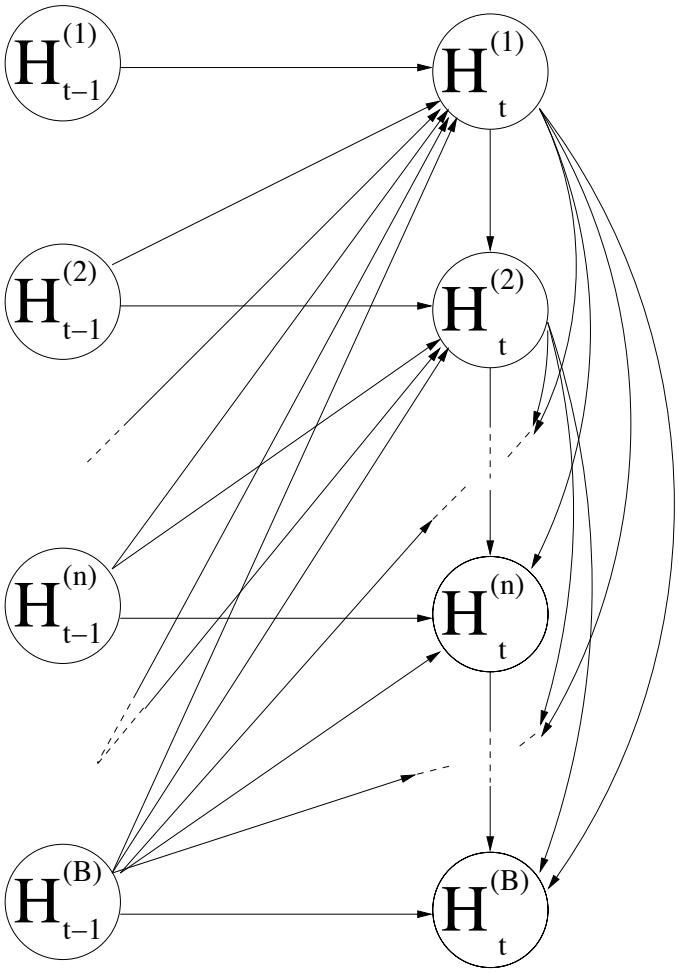

Figure 8: Time slice of a more complex multi-band DBN with the same inference complexity as an ergodic B-band DBN.

[10] K. Daoudi, D. Fohr, and C. Antoine. Continuous Multi-Band Speech Recognition using Bayesian Networks. In *IEEE Automatic Speech Recognition and Understanding Workshop (ASRU), Terento, Italy*, December 2001.

[11] A. Dawid. Applications of a general propagation algorithm for probabilistic expert systems. *Statistics and Computing*, (2):25–36, 1992.

[12] M. Deviren and K. Daoudi. Structural Learning of Dynamic Bayesian Networks in Speech Recognition. In *EUROSPEECH, Alborg, Denemark*, September 2001.

[13] M. Deviren and K. Daoudi. Continuous speech recognition using structural learning of dynamic Bayesian networks. In *EUSIPCO*, Toulouse, France, 2002.

[14] S. Dupont and H. Bourlard. Multiband approach for speech recognition. Workshop on Circuits, Systems, and Signal Processing 1996.

[15] H. Fletcher. *Speech and hearing in communication*. Krieger, New-York, 1953.

[16] Z. Ghahramani. An introduction to hidden markov models and bayesian networks. *International Journal of Pattern Recognition and Artificial Intelligence*, 15(1):9–42, 2001.

[17] G. Gravier. *Analyse statistique deux dimensions pour la modlisation segmentale du signal de parole: Application la reconnaissance.* PhD thesis, ENST Paris, 2000.

[18] G. Gravier, M. Sigelle, and G. Chollet. Markov Random Field modeling for Speech Recognition. *Australian Journal of Intelligent Information Processing Systems*, 5(4):245–252, 1999.

[19] D. Heckerman. A tutorial on learning with bayesian networks. Technical Report MSR-TR-95-06, Microsoft Research, Advanced Technology Division, March 1995.

[20] D. Heckerman, D. Geiger, and D.M. Chickering. Learning bayesian networks: The combination of knowledge and statistical data. *Machine Learning*, 20:197–243, 1995.

[21] H. Hermansky, S. Tibrewala, and M. Pavel. Towards ASR on partially corrupted speech. International Conference on Spoken Language Processing (ICSLP), 1996.

[22] F. Jensen, S. Lauritzen, and K. Olsen. Bayesian updating in recursive graphical models by local computations. *Computational Statistics and Data Analysis*, (4):269–282, 1990.

[23] M. Jordan, editor. Learning in graphical models. *MIT Press*, 1999.

[24] N. Mirghafori and N. Morgan. Transmissions and transitions: A study of two common assumptions in multi-band asr. In *International Conference on Acoustic, Speech and Signal Processing (ICASSP)*, 1998.

[25] N. Mirghafori and N. Morgan. Sooner or later: Exploring asynchrony in multi-band speech recognition. In *EUROSPEECH*, 1999.

[26] R.K. Moore. A dynamic programming algorithm for the distance between two finite areas. *IEEE Trans. on PAMI*, 1(1):86–88, 1979.

[27] A. Morris, A. Hagen, H. Glotin, and H. Bourlard. Multi-stream adaptive evidence combination for noise robust asr. *Speech Communication*, 2001.

[28] K. Murphy. *Dynamic Bayesian Networks: Representation, Inference and Learning.* PhD thesis, UC Berkeley, Computer Science Division, July 2002.

[29] J. Pearl. *Probabilistic reasoning in intelligent systems: Networks of plausible inference.* Morgan Kaufmann Publishers, San Mateo, CA, 1988.

[30] S. Lauritzen. Propagation of probabilities, means, and variances in mixed graphical association models. *Jour. Amer. Stat. Ass.*, 87(420):1098–1108, 1992.

[31] P. Smyth, D. Heckerman, and M.I. Jordan. Probabilistic independence networks for hidden Markov probability models. *Neural Computation*, 9(2):227–269, 1997.

[32] R. E. Tarjan and M. Yannakakis. Simple linear-time algorithms to test chordality of graphs, test acyclity of hypergraphs and selectively reduce acyclic hypergraphs. *SIAM Journal of Computing*, 13:566–579, 1984.

[33] K. Weber, S. Bengio, and H. Bourlard. Hmm2- extraction of formant features and their use for robust asr. In *EUROSPEECH*, 2001.

[34] K. Weber, S. Bengio, and H. Bourlard. Speech recognition using advanced hmm2 features. In *IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, 2001.

[35] K. Weber, S. Bengio, and H. Bourlard. Increasing speech recognition noise robustness with hmm2. In *International Conference on Acoustic, Speech and Signal Processing (ICASSP)*, 2002.

[36] M. Yannakakis. Computing fill-in is NP-complete. *SIAM Journal of Algebraic Discrete Mathods*, 2:77–79, 1981.

[37] G. G. Zweig and S. Russell. Probabilistic modeling with bayesian networks for automatic speech recognition. In *International Conference on Spoken Language Processing (ICSLP)*, Sydney, Australia, 1998.

[38] G. G. Zweig and S. Russell. Speech recognition with dynamic bayesian networks. In *Proceedings Fifteenth National Conference on Artificial Intelligence*, Madison, Wisconsin, 1998.

[39] G. G. Zweig and S. Russell. Probabilistic modeling with bayesian networks for automatic speech recognition. *Australian Journal of Intelligent Information Processing Systems*, 5(4):253–260, 1999.

[40] G.G. Zweig. *Speech Recognition with Dynamic Bayesian Networks*. PhD thesis, University of California, Berkeley, 1998.