A conditional model for automatic orchestration Application to a real-time Live Orchestral Piano

LEOPOLD CRESTEL, Institut de Recherche et Coordination Acoustique/Musique
PHILIPPE ESLING, Institut de Recherche et Coordination Acoustique/Musique

Multifrequency media access control has been well understood in general wireless ad hoc networks, while in wireless sensor networks, researchers still focus on single frequency solutions. In wireless sensor networks, each device is typically equipped with a single radio transceiver and applications adopt much smaller packet sizes compared to those in general wireless ad hoc networks. Hence, the multifrequency MAC protocols proposed for general wireless ad hoc networks are not suitable for wireless sensor network applications, which we further demonstrate through our simulation experiments. In this article, we propose MMSN, which takes advantage of multifrequency availability while, at the same time, takes into consideration the restrictions of wireless sensor networks. Through extensive experiments, MMSN exhibits the prominent ability to utilize parallel transmissions among neighboring nodes. When multiple physical frequencies are available, it also achieves increased energy efficiency, demonstrating the ability to work against radio interference and the tolerance to a wide range of measured time synchronization errors.

Additional Key Words and Phrases: Automatic orchestration, Real-time, Conditional Restricted Boltzmann Machine, time modeling

ACM Reference Format:
Léopold Crestel, Philippe Esling, 2010. A conditional model for automatic orchestration
Application to a real-time Live Orchestral Piano. *ACM Trans. Embedd. Comput. Syst.* V, N, Article A (January YYYY), 10 pages.
DOI: http://dx.doi.org/10.1145/0000000.0000000

1. INTRODUCTION

Musical orchestration is the subtle art of of writing musical pieces for orchestra, by combining the spectral properties specific to each instrument in order to achieve a particular sonic goal. This complex discipline involves a wide set of intricate mechanisms, most of which have not yet been satisfactorily theorized. Indeed, famous composers often conjectured that orchestration would mainly remain an empirical discipline, which could only be learned through experience and never axiomatized in books. Even if several famous musicians have written orchestration treatises [Berlioz 1844; Koechlin 1941], those mostly remain recommendations and sets of existing orchestration examples from which one can draw inspiration. We focus more specifically in this work on *projective* orchestration, which is the transformation from a piano score to an orchestral piece. Many composers have worked in a projective manner, and a large amount of example can be found in the repertoire. For instance, one of the most famous is the orchestration of *Les tableaux d'une exposition*, a Modest Moussorgsky's piano piece, by Maurice Ravel.

This work is supported by the National Science Foundation, under grant CNS-0435060, grant CCR-0325197 and grant EN-CS-0329609.
Author's addresses: L. Crestel and P.Esling, Représentation Musicales, Institut de Recherche et Coordination Acoustique/Musique;

DOI: http://dx.doi.org/10.1145/0000000.0000000

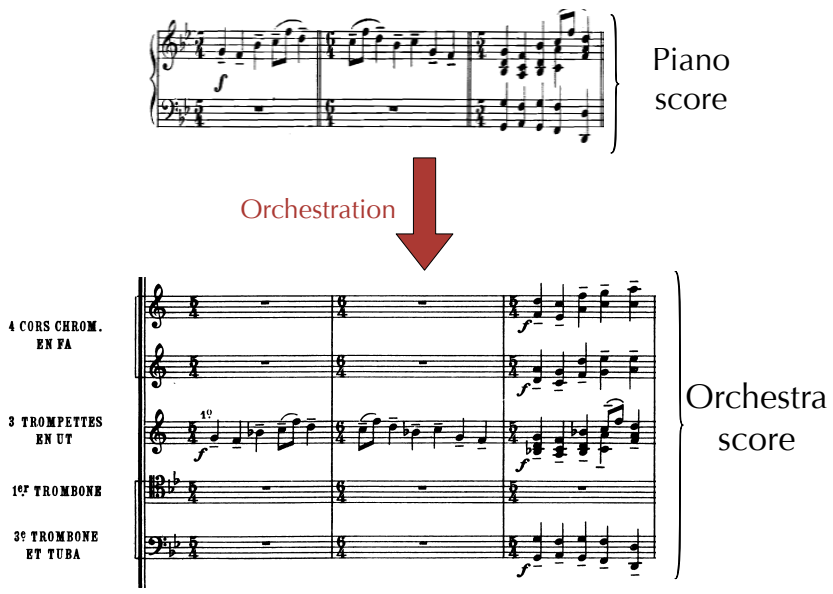


Fig. 1. *Projective orchestration*. A piano score is extended (projected) on an orchestra. For one piano score, many acceptable orchestration exist. Our hypothesis is that a piano score is strongly correlated to any of the orchestration that could be produced from this piece.

The objective in this work is to be able to automatically perform in real-time the *projective* orchestration of a piano performance. More specifically, our system takes in input a piano score and outputs an orchestral score. The vast combinatorial set of instrument possibilities added to the complex temporal structure of polyphonic music make this problem a particularly daunting task. Several attempts to build an automatic orchestration system can be found in the literature. Orchestration can be viewed as assigning the different notes of the piano score to a certain number of instrument according to constraints over the number of instruments, their tessitura, a certain voice leading inside an instrument. Interpreting orchestration as a Constraint Solving Problem (CSP) lead to a first solution [Truchet and Assayag 2011]. However, as Steven McAdams pointed it out, timbre is "a structuring force in music" [McAdams 2013] in the sense that it should be used to emphasize the already existing movements of the original piano piece. We believe that a system only built only on symbolic constraint will undoubtedly fail at grasping this underlying structure. Hence, orchestrating first require to understand the harmonic, rhythmic and melodic structure of the original piano piece. *Orchids* ([Esling et al. 2010]) is an other interesting work set in an other paradigm called *injective* orchestration. It consists in trying to reconstruct a target timbre for a small temporal frame. The major drawback being that the orchestration is limited to short (less than 10 seconds) examples. In order to build an automatic orchestration system being able to work on a macro-temporal timescale while structuring the musical discourse, statistical inference appeared us to be a promising solution. Their should indeed exist strong correlation between the information contained in the original piano score and the orchestral rendering we want to produce. Statistical inference would allow us to extract the knowledge and rules embodied in the many orchestration proposed by famous composers over the years.

We decided to work with a class of models called conditional models [Taylor 2009], which derive from a particular type of Markov Random Field called the Restricted

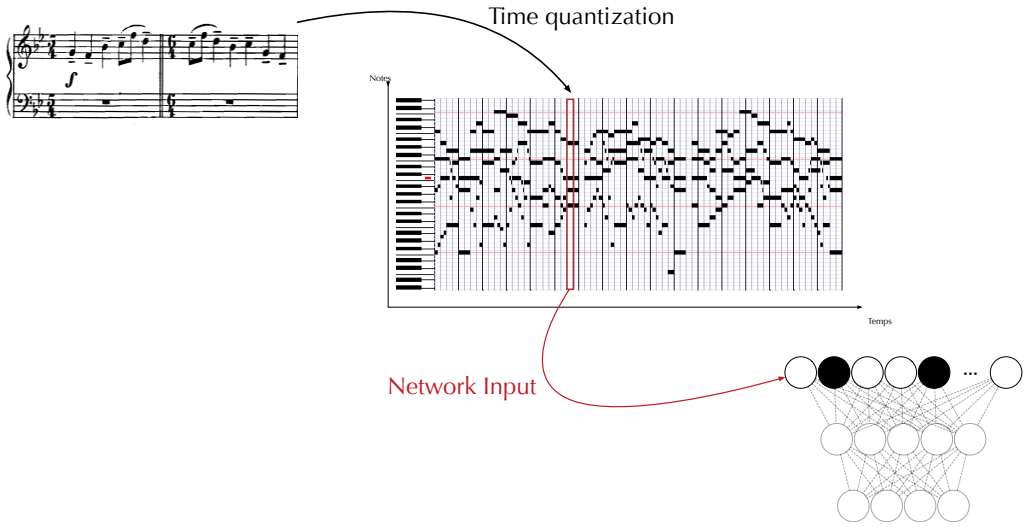


Fig. 2. *Data representation for a single piano*. The pianoroll is a representation of musical events, discrete on both frequency scale (pitch) and the time scale (frames). A pitch p at time t can be either on or off, which is represented by a one or a zero on the pianoroll. To represent an orchestra, the pianorolls of each instruments are simply concatenated along the pitch dimension.

Boltzmann Machine (RBM) [Fischer and Igel 2014]. While being able to model complex distributions through latent units, those models implement a notion of context which is interesting in our case in order to model the influence of the past over the present and of the piano over the orchestra. Those models are generative which is a requirement in our case. If correctly trained on a training dataset, a model then has the ability to generate data that are similar, yet unseen, to those contained in the training set. This is through this mechanism that orchestral inference can be performed.

We propose in this article a new evaluation framework for the orchestral inference task in order to evaluate the different model we proposed. Building a quantitative evaluation framework for generative models is rarely straightforward, especially since computing the likelihood of a test sample is intractable in the model we used. A common practice is to define an auxiliary task. Hence, we rely on a frame-level predictive task based on an accuracy measure which basically consists in comparing, for a given piano score, the orchestration proposed by our model and an orchestration written by a composer. The results of the proposed model are then presented. We picked out the best model and included it in a real-time orchestration system called *LOP*.

This paper is organized as follows. In sections 2 we introduce the state of the art in conditional models through three well known models: the RBM, the CRBM and the FGCRBM. The orchestration inference task is presented in the section 3 along with an evaluation framework based on a frame-level accuracy measure. The previously introduced models are then evaluated in this framework and the results displayed. The section 4 introduces a real-time *projective* orchestration system using the presented architectures.

2. STATE OF THE ART

2.1. Data representation

To model sequences of symbolic music, the pianoroll representation is often used. A pianoroll is a matrix whose rows and columns are a discretisation of pitch and time figure

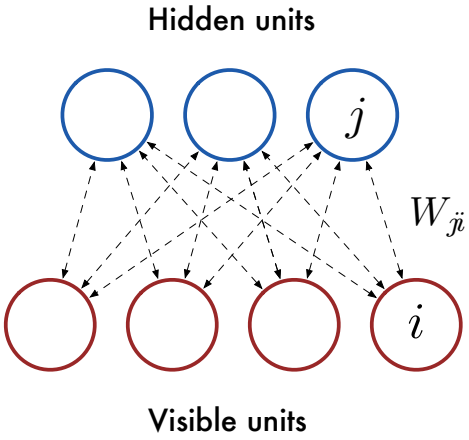


Fig. 3. *Restricted Boltzmann Machine*. The RBM is an energy-based model. Its energy function is computed from the values of weights that link nodes. Training an RBM consists in lowering the energy function around the example from a training set. Inference in this model is easy to perform since the hidden (resp. visible) units are independent from each others.

2 on page 3. Note that this discretisation flows naturally from the scores notation in western music since notes are aligned on a discrete pitch scale and rhythmically on the beat. The pitch p being played at time t is then represented in the pianoroll representation by $Pianoroll(p,t) = 1$, $Pianoroll(p,t) = 0$ meaning that pitch p is not played at time t. The dynamics are ignored and each time frame is a binary vector that indicates either a pitch is on or off. This representation, usually defined for a single polyphonic instrument can easily be extended to an orchestra composed by N instruments by simply concatenating the pianoroll of each instrument over the pitch dimension. Note that we respect the usual simplifications used when writing orchestral scores which consists in grouping all the instruments of a same section. For instance, the section *violin 1*, composed by many instrumentalists (about 10), is represented as a unique instrument.

2.2. Restricted-Boltzmann Machine

A Restricted-Boltzmann Machine (RBM) [Hinton et al. 2006] is an energy-based model that represent the joint distribution of a visible vector $\mathbf{v} = (v_1, ..., v_m)$ and a hidden vector $\mathbf{h} = (h_1, ..., h_n)$. This distribution is given by $p(v,h) = \frac{\exp^{-E(v,h)}}{Z}$ where

$$E(\mathbf{v}, \mathbf{h}) = -\sum_{i=1}^{m} a_i v_i - \sum_{j=1}^{n} b_j h_j - \sum_{i=1}^{m}\sum_{j=1}^{n} v_i W_{ij} h_j \tag{1}$$

and $Z = \sum_{v,h} \exp^{-E(v,h)}$ is a usually intractable partition function. $\Theta = \{W, a, b\}$ are the weights of the network. Each unit represent an activation function which is a simple non-linear function. Unfortunately, the gradient of the negative log-likelihood of a vector from the training database $\mathbf{v}^{(l)}$ is intractable. A training algorithm called Contrastive divergence (CD) [Hinton 2002] rely on an approximation of the model driven

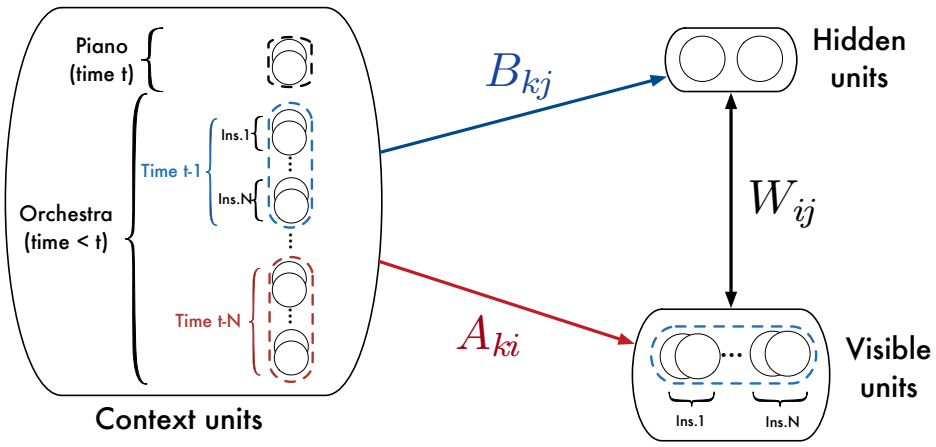


Fig. 4. *Conditional RBM*. A layer of context units is added to the standard RBM architectures. Those context units linearly modify the bias of both visible and hidden units.

term of this equation by running a k-step Gibbs chain to obtain a sample $v^{(l,k)}$ ((2))

$$-\frac{\partial \ln(p(\boldsymbol{v^{(l)}}|\boldsymbol{\Theta}))}{\partial \boldsymbol{\Theta}} = \mathbb{E}_{p(\mathbf{h}|\boldsymbol{v^{(l)}})}\left[\frac{\partial E(\boldsymbol{v^{(l)}},\mathbf{h})}{\partial \boldsymbol{\Theta}}\right] - \mathbb{E}_{p(\mathbf{h},\mathbf{v})}\left[\frac{\partial E(\mathbf{v},\mathbf{h})}{\partial \boldsymbol{\Theta}}\right] \tag{2}$$

$$\approx \mathbb{E}_{p(\mathbf{h}|\boldsymbol{v^{(l)}})}\left[\frac{\partial E(\boldsymbol{v^{(l)}},\mathbf{h})}{\partial \boldsymbol{\Theta}}\right] - \mathbb{E}_{p(\mathbf{h}|\boldsymbol{v^{(l,k)}})}\left[\frac{\partial E(\boldsymbol{v^{(l,k)}},\mathbf{h})}{\partial \boldsymbol{\Theta}}\right] \tag{3}$$

Running a Gibbs sampling chain consists in alternatively sampling the hidden units knowing the visible units then the visible units knowing the inferred hidden units by using the marginal probabilities ((4)).

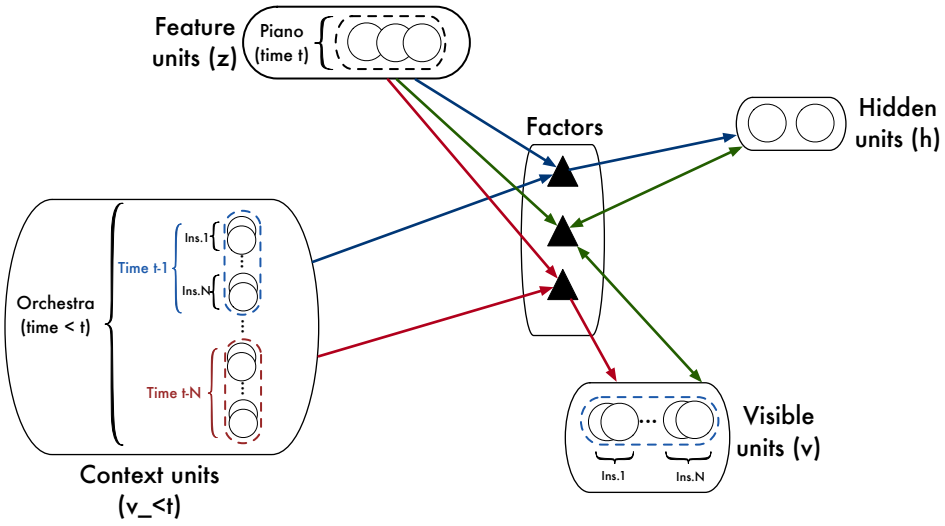
$$p(v_i = 1|\mathbf{h}) = sigm\left(b_i^{(v)} + \sum_j W_{ij} h_j\right) \tag{4}$$

$$p(h_j = 1|\mathbf{v}) = sigm\left(b_j^{(h)} + \sum_i W_{ij} v_i\right) \tag{5}$$

where $sigm$ is the sigmoid function. It has been proved that the samples we obtain after an infinite number of iteration will be drawn from the joint distribution of the visible and hidden units of our model. An other approximation consists in starting the Gibbs chain from the sample $v^{(l)}$, which increases the convergence of the chain, and to limit the number of alternate sampling steps to a fixed number K, which leads to the CD-K algorithm used to train a RBM.

2.3. Conditional RBM

The Conditional Restricted Boltzmann Machine (CRBM) model ([Taylor 2009]) is a standard RBM in which a dynamic biases conditioned is added to the visible and hidden units. The dynamic bias linearly depends of context units x. To model time series, if we consider that the visible units represent the current time frame, those context units can be defined as the concatenation of the N last time frames. We call this vector



$\boldsymbol{v}_{k<t} = \left(v_1^{(t-N)}, ..., v_m^{(t-N)}, ..., v_1^{(t)} ..., v_m^{(t)}\right)$, where N denotes the order of the model. The energy function of the Conditional RBM is given by ((6))

$$E(v_t, h_t|v_{<t}) = -\sum_i \hat{a}_{i,t} v_{i,t} - \sum_{ij} W_{ij} v_{i,t} h_{j,t} - \sum_j \hat{b}_{j,t} h_{j,t} \tag{6}$$

where the biases are defined by $\hat{a}_i = a_i + \sum_k A_{ki} v_{k,<t}$ and $\hat{b}_j = b_j + \sum_k B_{kj} v_{k,<t}$.

If we consider that the visible units represent the orchestral vector for the time frame t, conditional units can be used to model the influence of the past orchestral vector $t-1, ..., t-N$ (the concatenation of those N orchestral frames are then referred to as past orchestral vector) and the (strong) influence of the piano frame at time t over the visible units. In the CRBM model, the conditional units are the concatenation of the past orchestral vector and the current piano vector.

This model appeared as an interesting solution in order to model the strong temporal relations underlying in symbolic music data. It can be trained by contrastive divergence, since the marginal probability of visible and hidden units can be easily obtained from the energy distribution.

2.4. Factored Gated Conditional RBM

The Factored Gated Conditional RBM model [Taylor and Hinton 2009] proposes to extend the Conditional RBM model by adding a layer of feature unit z_l which modulate the weights of the conditional architecture in a multiplicative way. Hence, the weights of the networks become $\Theta = \left\{W_{ijl}, A_{ikl}, B_{jkl}, \hat{a}_i, \hat{b}_j\right\}$. This multiplicative influence can be understood as a modification of the energy landscape of the model. Since the number of parameter to train becomes high, the 3 dimensional tensors can be factorized into a product of three matrices by including factors $W_{ijl} = W_{if}.W_{jf}.W_{lf}$. The energy function of this Factored Gated Conditional RBM is then given by

$$E(v_t, h_t|v_{<t}, y_t) = -\sum_f \sum_{ijl} W_{if}^v W_{jf}^h W_{lf}^z v_{i,t} h_{j,t} z_{l,t} - \sum_i \hat{a}_{i,t} v_{i,t} - \sum_j \hat{b}_{j,t} h_{j,t} \tag{7}$$

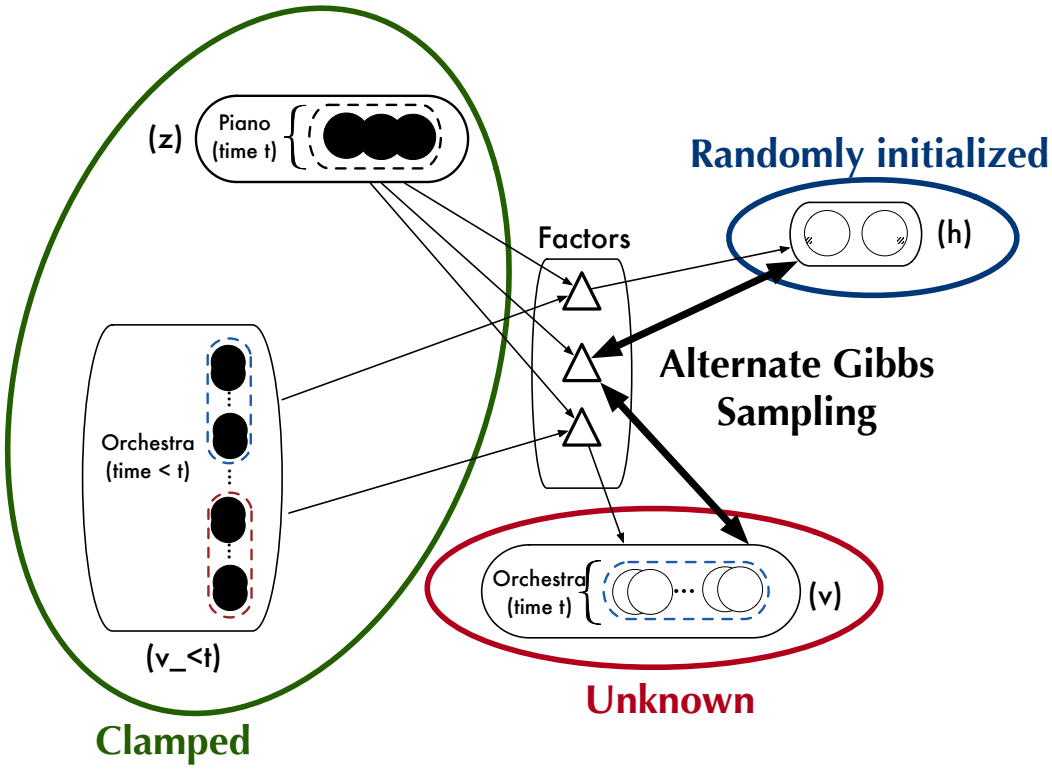


Fig. 5. *Sampling in a FGCRBM*. Context and Features units are respectively clamped to the last ($t-1$ to $t-N$) orchestral frames and the current (t) piano frame. Hidden units are randomly initialized. Then, several Gibbs sampling step are performed, typically 50.

where the dynamic biases of the visible and hidden units are defined by

$$\hat{a}_{i,t} = a_i + \sum_m \sum_{kl} A^v_{im} A^{v<t}_{km} A^z_{lm} v_{k,<t} z_{l,t} \tag{8}$$

$$\hat{b}_{j,t} = b_j + \sum_n \sum_{kl} B^h_{jn} B^{v<t}_{kn} B^z_{ln} v_{k,<t} z_{l,t} \tag{9}$$

The FGCRBM offers the possibility to distinguish between the influence of the current piano vector and the past orchestral vectors by assigning the first one to the features units z_l and the second to the context units x_k.

2.5. Generative models

The previously introduced models are generative models. After the training phase, the distribution represented by the networks is supposed to be close to the underlying distribution of the data. It is then possible to sample from this distribution to reproduce data that are alike the data of the training set. Conditional models allow to impose a certain context, which enables to generate sequences of data under a certain context. We remind that our objective is to transform a sequence of binary vectors drawn from a piano score (referred to as piano vectors) into a sequence of orchestral vectors (section 2.1).

Knowing the current piano frame and the recent past orchestral frames, the generation process in a conditional model can be described as follow. After randomly setting the hidden units, a certain number (typically 50) of Gibbs sampling steps are performed in order to reach the equilibrium distribution of the model. This process is described for the FGCRBM model in figure 5 on page 7. This task correspond exactly to the projection of the piano

3. PROJECTIVE ORCHESTRATION

A projective orchestration task is introduced in this section, along with an evaluation framework and the performances of our model in this framework. The evaluation is a frame-level prediction task that rely on an accuracy measure.

3.1. Formalization

Automatic orchestration suffers from the lack of quantitative evaluation. The different work on the domain mainly rely on qualitative evaluation [Handelman et al. 2012]. To our best knowledge, there has not been any attempt in the automatic orchestration field to define a task associated to a performance measure. We propose here a first attempt in order to fill this gap by defining the orchestration prediction task.

An orchestral pianoroll is defined by the concatenation of the instrumental pianoroll

$$Orch = \begin{bmatrix} Instrument1 \\ Instrument2 \\ \vdots \\ InstrumentN \end{bmatrix} \tag{10}$$

The dimension of a matrix $M(t)$ is the number of instrument per the number of pitch : $N_{instrument} \times 88$. We reduce the number of possible pitch to those of a grand piano (88 pitch from). One can argue that the ambitus of a piccolo for instance goes higher than the highest note of a piano in frequency. Since we work only with symbolic notations, the symbolic score of a piccolo is comprised in the symbolic score of a piano (the piccolo play the note written on the score an octave higher). In our framework we chose 14 instruments indexed by :

1. Violin
2. Viola
3. Cello
4. Double-bass
5. Harp
6. Timpani
7. Trumpet
8. Trombone
9. Tuba
10. French or Eng. horn
11. Oboe
12. Bassoon
13. Clarinet
14.Flute

3.2. Evaluation

3.2.1. Frame-level accuracy. For each line of the matrix, i.e. each instrument, we can compute the frame-level accuracy as presented in the previous section, and then sum the accuracy of all the instrument

$$Acc_{orchestral} = \sum_{i=1}^{N_{instrument}} Acc(M(i,t)) \tag{11}$$

where $M(i,t)$ is 88 size vector constituted by the pitches of the instrument i. Note that in order to compare different models, it is necessary to have the same number of instruments $N_instruments$.

3.2.2. Event-level accuracy.

3.3. Database

We used a parallel database of piano scores and their orchestration by famous composers. The database consists of 76 *XML* files. Given the complexity of the distribution we wanted to model and the reduced size of the fatabase we have accessed to, we decided to keep as a test dataset only the last half of one track from our database. Hence 75 and a half files were used to train our model. We chose to do so in order to have the best generation ability. For each instrument, the pitch range is reduced to the tessitura observed in the training dataset. We used a rhythmic quantization of 8 frame per beat, which means that the smallest symbolic rhythm is a 32^{th} note.

3.4. Results

4. LIVE ORCHESTRAL PIANO

5. CONCLUSION AND FUTURE WORKS

6. TYPICAL REFERENCES IN NEW ACM REFERENCE FORMAT

A paginated journal article [**?**], an enumerated journal article [**?**], a reference to an entire issue [**?**], a monograph (whole book) [**?**], a monograph/whole book in a series (see 2a in spec. document) [**?**], a divisible-book such as an anthology or compilation [**?**] followed by the same example, however we only output the series if the volume number is given [**?**] (so Editor00a's series should NOT be present since it has no vol. no.), a chapter in a divisible book [**?**], a chapter in a divisible book in a series [**?**], a multi-volume work as book [**?**], an article in a proceedings (of a conference, symposium, workshop for example) (paginated proceedings article) [**?**], a proceedings article with all possible elements [**?**], an example of an enumerated proceedings article [**?**], an informally published work [**?**], a doctoral dissertation [**?**], a master's thesis: [**?**], an online document / world wide web resource [**?**], [**?**], [**?**], a video game (Case 1) [**?**] and (Case 2) [**?**] and [**?**] and (Case 3) a patent [**?**], work accepted for publication [**?**], 'YYYYb'-test for prolific author [**?**] and [**?**]. Other cites might contain 'duplicate' DOI and URLs (some SIAM articles) [**?**]. Boris / Barbara Beeton: multi-volume works as books [**?**] and [**?**].

APPENDIX

In this appendix, we measure the channel switching time of Micaz [CROSSBOW] sensor devices. In our experiments, one mote alternatingly switches between Channels 11 and 12. Every time after the node switches to a channel, it sends out a packet immediately and then changes to a new channel as soon as the transmission is finished. We measure the number of packets the test mote can send in 10 seconds, denoted as N_1. In contrast, we also measure the same value of the test mote without switching channels, denoted as N_2. We calculate the channel-switching time s as

$$s = \frac{10}{N_1} - \frac{10}{N_2}.$$

By repeating the experiments 100 times, we get the average channel-switching time of Micaz motes: 24.3μs.

ELECTRONIC APPENDIX

The electronic appendix for this article can be accessed in the ACM Digital Library.

ACKNOWLEDGMENTS

The authors would like to thank Dr. Maura Turolla of Telecom Italia for providing specifications about the application scenario.

REFERENCES

Hector Berlioz. 1844. *Grand traité d'instrumentation et d'orchestration modernes*. Schonenberger.

Philippe Esling, Grégoire Carpentier, and Carlos Agon. 2010. Dynamic Musical Orchestration Using Genetic Algorithms and a Spectro-Temporal Description of Musical Instruments. *Applications of Evolutionary Computation* (2010), 371–380.

Asja Fischer and Christian Igel. 2014. Training restricted Boltzmann machines: An introduction. *Pattern Recognition* 47, 1 (2014), 25–39.

Eliot Handelman, Andie Sigler, and David Donna. 2012. Automatic orchestration for automatic composition. *In Proc. of MUME* (2012), 43–48.

Geoffrey E Hinton. 2002. Training products of experts by minimizing contrastive divergence. *Neural computation* 14, 8 (2002), 1771–1800.

Geoffrey E. Hinton, Simon Osindero, and Yee-Whye Teh. 2006. A Fast Learning Algorithm for Deep Belief Nets. *Neural Comput.* 18, 7 (July 2006), 1527–1554. DOI:http://dx.doi.org/10.1162/neco.2006.18.7.1527

Charles Koechlin. 1941. *Traité de l'orchestration*. Éditions Max Eschig.

Stephen McAdams. 2013. Timbre as a structuring force in music. In *Proceedings of Meetings on Acoustics*, Vol. 19. Acoustical Society of America, 035050.

Graham William Taylor. 2009. *Composable, distributed-state models for high-dimensional time series*. Ph.D. Dissertation. University of Toronto.

Graham W Taylor and Geoffrey E Hinton. 2009. Factored conditional restricted Boltzmann machines for modeling motion style. In *Proceedings of the 26th annual international conference on machine learning*. ACM, 1025–1032.

Charlotte Truchet and Gerard Assayag. 2011. *Constraint Programming in Music*. Wiley.

Received February 2007; revised March 2009; accepted June 2009

Online Appendix to: A conditional model for automatic orchestration Application to a real-time Live Orchestral Piano

LEOPOLD CRESTEL, Institut de Recherche et Coordination Acoustique/Musique
PHILIPPE ESLING, Institut de Recherche et Coordination Acoustique/Musique

A. THIS IS AN EXAMPLE OF APPENDIX SECTION HEAD

Channel-switching time is measured as the time length it takes for motes to successfully switch from one channel to another. This parameter impacts the maximum network throughput, because motes cannot receive or send any packet during this period of time, and it also affects the efficiency of toggle snooping in MMSN, where motes need to sense through channels rapidly.

By repeating experiments 100 times, we get the average channel-switching time of Micaz motes: 24.3 μs. We then conduct the same experiments with different Micaz motes, as well as experiments with the transmitter switching from Channel 11 to other channels. In both scenarios, the channel-switching time does not have obvious changes. (In our experiments, all values are in the range of 23.6 μs to 24.9 μs.)

B. APPENDIX SECTION HEAD

The primary consumer of energy in WSNs is idle listening. The key to reduce idle listening is executing low duty-cycle on nodes. Two primary approaches are considered in controlling duty-cycles in the MAC layer.

DOI: http://dx.doi.org/10.1145/0000000.0000000